REVISTA DE INFORMÁTICA • ABRIL 1987 • 300 ESC.

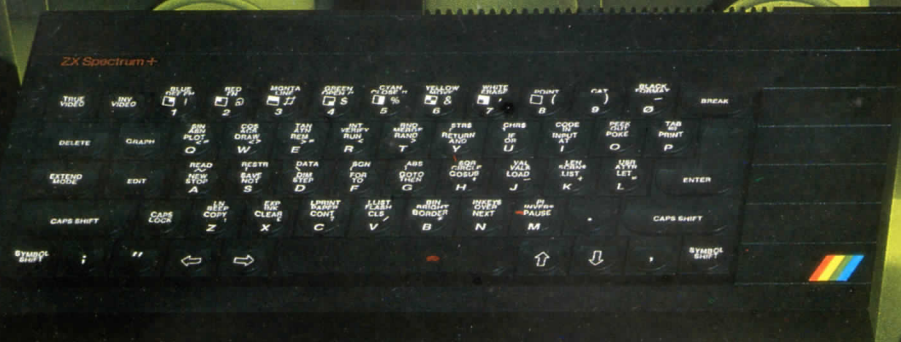
CONCURSO

ROTINA PARA ESCREVER NA PISTA 04 (DIRECTORIA DA DISKETTE)

MANUAL TÉCNICO DAS FLOPPY DISK DRIVE TIMEX

BETA BASIC 3.0

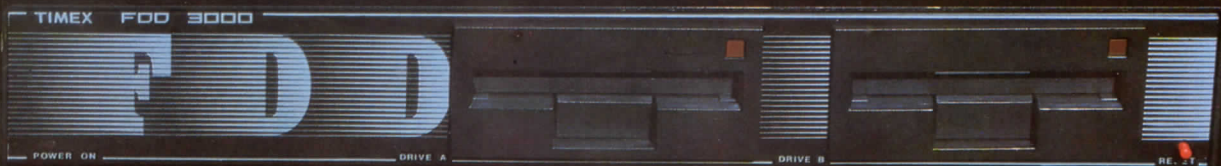
POSTER «FAIRLIGHT»

ZX SPECTRUM 128K

O *TIMEX FDD 3000* é um sistema de discos com sistema operativo próprio (TIMEX OPERATING SYSTEM) T.O.S.

Permite também utilizar o sistema operativo *CP/M, DA DIGITAL RESEARCH*, com o *SPECTRUM 48K, SPECTRUM PLUS, TC 2048, e TC 2068.*

CP/M é o mais popular sistema operativo de 8 bits com aplicações em áreas, tais como: Negócios, Processamento de Texto, Finanças, Advocacia, Gestão, Base de Dados, Linguagens e compiladores. *CP/M* abre a porta a uma biblioteca de mais de *15 000 programas.*

T.O.S. É UM SISTEMA POTENTE

— Utiliza "Keywords" como uma extensão ao *SINCLAIR BASIC.*
— Nem um só byte de memória do computador é utilizado.
— Permite "PRINT" e "INPUT" através dos portos RS 232 usando "KEYWORDS" em programas BASIC.
— Permite acesso a ficheiros Sequencial e/ou Aleatório — "RANDOM" — até 16 canais.
— Possui uma poderosa estrutura de directorios em árvore.
— Suporta todos os tipos de "DATA" do Spectrum e permite outros.

TIMEX

TIMEX

CAMPANHA DE LANÇAMENTO

Durante a campanha de lançamento é válida a troca do seu TIMEX FDD pelo novo TIMEX FDD 3000 (com dois disk drives) e SISTEMA OPERATIVO CP/M *APENAS PELO CUSTO DE UM DRIVE B.*

** SOFTWARE T.O.S.*

— TASWORD II (Processador de texto em TOS)
— ALFACONT (Contabilidade Geral — P.O.C.)
— GESTIN I (Gestão integrada de Stocks da PROCOMPE)
— AGENDA TELEFÓNICA
— AGENDA DIÁRIA
— GRAFLOG (Programa de Gráficos)
— GESTIVA (Controlo de I.V.A)

** SOFTWARE CP/M*

PASCAL (Linguagem)
C++ (Linguagem)
DEVPAC (Assembler)

FLEXIWRITE (Processador de texto)
FLEXICALC (Folha de cálculo)
FLEXIFILE (Base de dados)

SOFTWARE PORTUGUÊS

SOFTFILE *Software*

Sumário

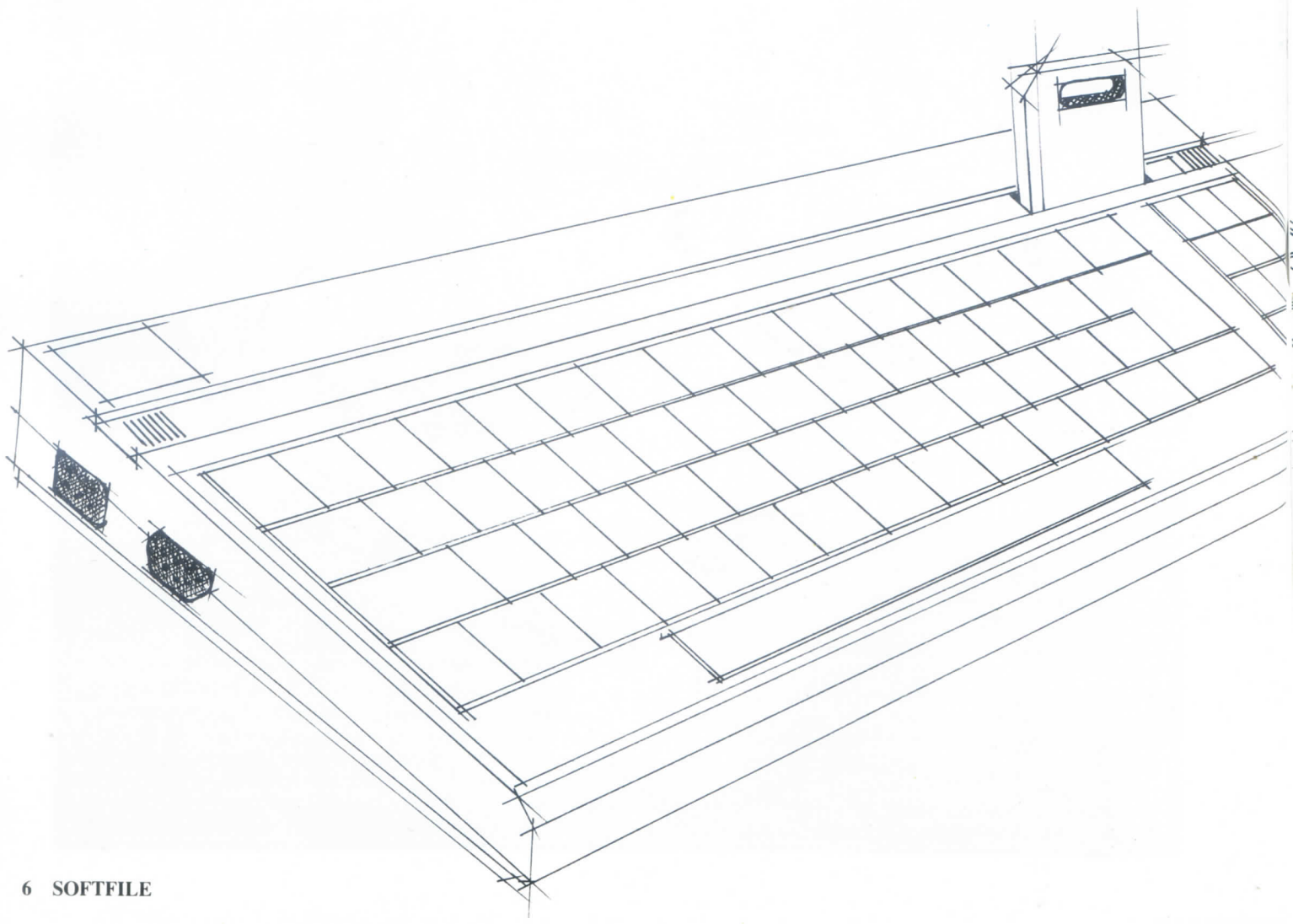
REVISTA SOFTFILE - R. Rodrigo da Fonseca, 95-4º, 1200 Lisboa - Tel. 65 90 47 / 68 40 22 - Telex 93011 VAGUI P • **Editor** - Manuel Bravo • **Director** - Horácio Mariano • **Colaboradores** - Abílio Pereira, Armando da Silva, Carlos Filipe, João Varela, Paulo Bernardo • **Publicidade** - Carlos Aguda • **Assinaturas** - José Vieira • **Maquetagem** - Carlos Aguda • **Impressão/Montagem** - ROLIMPRE, Artes Gráficas, Lda. • **Distribuidora** - MIDESA - Marlo Ibéria Distribuição de Edições, S.A. • A revista SOFTFILE está inscrita na Direcção Geral de Comunicação Social com o número 111166 • Depósito Legal: 10330/85 • • **Tiragem** 15 000 exemplares.

NOTÍ

TIMEX - NOVOS PRODUTOS

Quem pensar que a TIMEX (TMX PORTUGAL, LTD) parou ou está com dificuldades no campo da criação enganou-se e redondamente.

Segundo nos foi dado observar e comunicado os produtos em questão o TC 3256, 256K Personal Computer, e o gravador estarão a público proximamente.

Embora já operacionais a sua saída está atrasada pela dificuldade na aquisição de determinados componentes no mercado internacional (prazos de entrega extremamente

longos) o que provoca, obviamente, atrasos no arranque de produção.

O novo computador TIMEX TC 3256 é um personal computer de excelente concepção e características inovadoras.

A nova máquina da TIMEX possui à partida 256K RAM podendo ser expansível até 1 Mbyte. A CPU é um Z80A.

Ao ligá-lo o utilizador tem à sua frente, no ecrã, cinco opções que definem a sua estrutura base: BASIC, TIMEWORD+, CP/M, TERMINAL, DISK START e CARTRIDGE.

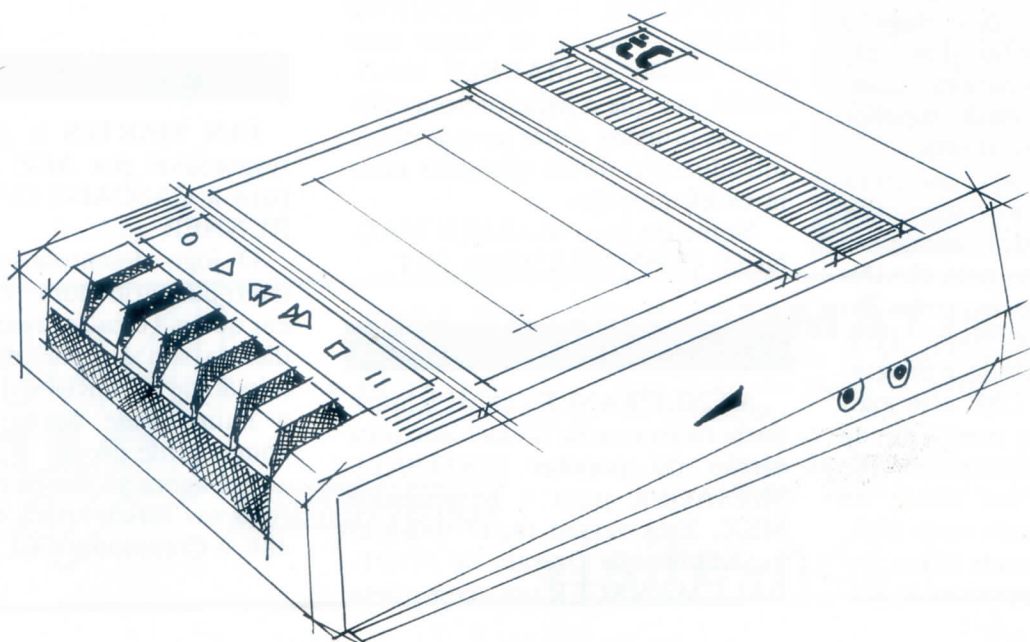
O «Basic» é o Basic da linha TIMEX (SPECTRUM compatível) mas agora com novos comandos. O «TIMEWORD+» é um processador de texto que, sinceramente, quem conhece o WORDSTAR ficará confuso ao verificar a sua similaridade e ainda com a possibilidade de inclusão de screens no próprio texto. A opção «CP/M Terminal» para utilização com o sistema FDD 3000 e consequentemente em CP/M. «DISK START» é a opção com maior interesse e inovadora porquanto não é oferecida aos utilizadores em nenhum micro do mercado desta categoria. esta opção reserva 64K de memória e permite ao utilizador «expert» criar ou introduzir a linguagem que entenda. Parabéns pela ideia.

A última opção é a denominada «CARTRIDGE» que como o seu nome indica baseia-se na utilização de programas em cartridge.

O aspecto exterior do TC 3256 é de louvar apresentando um teclado tipo profissional com parte de numérica. Possui várias saídas: JOYSTICK RS 232-C, BUS EXPANSÃO, PDD, I/O BUS, RGB, EAR/MIC, TV TENET (rede de utilizadores).

A nível de resolução dispõe de dois modos, ambos seleccionáveis pelo operador, os de 256x176 e 512x176 pixels.

O gravador «TIMEX COMPUTER PROGRAM RECORDER» já não tão elegante pareceu-nos também um bom item.

Pensamos, na verdade, que estes novos produtos saídos dos «cérebros» da TIMEX irão constituir um enorme sucesso não só em Portugal mas também no estrangeiro.

NOTÍCIAS

RAINBIRD SOFTWARE

A conhecida firma de SOFTWARE RAINBIRD lançou recentemente no mercado dois novos jogos de acção.

TRACKER trata-se de um excelente jogo a três dimensões aliando acção e estratégia, a acção desenrola-se num tunel tridimensional numa batalha táctica contra um andróide dotado de inteligência artificial.

O objectivo do jogo consiste em alcançar o quartel general de comunicações do andróide situado no sector central da fortaleza. Para os fans de «Arcade Game», STARGLIDER irá constituir sem dúvida um sucesso. Starglider é uma nave em missão em planetas distantes atacando naves inimigas e passando os vários obstáculos que lhe dificultam a missão, até aqui nada de novo, o grande trunfo deste jogo é representado pelos excelentes gráficos e os extraordinários efeitos sonoros, inclusivamente onde uma voz feminina o informa de como está a decorrer a missão.

STARGLIDER está disponível em versões para o ATARY ST, os TIMEX/SPECTRUM 48K e o SPECTRUM 128K constituindo quanto a nós o melhor jogo até agora comercializado para este computador e sendo uma versão superior à comercializada para o 48K.

Contamos num dos próximos números apresentar um artigo mais detalhado sobre estes excelentes jogos agora lançados pela RAINBIRD. Na linha do bem conhecido Ditador, surgiu este Natal THE PRESIDENT, concebido e programado por KEVIN TOM, este jogo coloca-o no lugar do presidente de um país imaginário concedendo-lhe poder absoluto. A sua tarefa no entanto não se apresenta nada fácil, pois terá simultaneamente de enfrentar forças inimigas, tentativas de golpes de Estado, aumentar a produção, combater a crise e a inflação e lutar para ser novamente reeleito.

MASTERTRONICS SOFTWARE

Baseando-se numa famosa banda desenhada a MASTERTRONICS apresenta FLASH GORDON em luta com o império do ditador MING.

O jogo desenrola-se em três níveis, no primeiro FLASH GORDON tem que atravessar a selva que rodeia o palácio de MING e encontrar-se com o príncipe BARIN, tarefa nada fácil pois a selva apresenta-se repleta de estranhos e perigosos habitantes.

No segundo nível Flash combate com o príncipe e tenta derrotá-lo de modo a ter acesso ao palácio de Ming. O terceiro e último nível é talvez o mais excitante Flash encontra-se finalmente com Ming com quem tem de combater.

MELBOURNE HOUSE SOFTWARE

A mesma equipa que criou GYROSCOPE — MELBOURNE HOUSE — acaba de lançar uma nova versão de MARBLE MADNESS, em que o jogador pode criar as suas próprias pistas para as bolas, introduzindo novos aliciantes neste já excelente jogo.

Nome do jogo MARBLE MADNESS CONSTRUCTION SET.

ENDURANCE GAMES

A ENDURANCE GAMES introduziu recentemente no mercado uma versão do popular FOOTBALL MANAGER para o computador MSX. Esta versão inclui todas as qualidades que fizeram de FOOTBALL MANAGER um «best-seller» para outros computadores. Estão pois de parabéns os utilizadores de MSX.

ARGUS PRESS SOFTWARE

Uma das séries mais populares da BBC GRANGE HILL, baseada no livro «GRANGE HILL AFTER HOURS», está agora a ser transformada num jogo para computador pela ARGUS PRESS SOFTWARE.

A acção decorre numa escola onde o personagem «GONCH» (LUKE GARDNER), controlada pelo jogador, tem que entrar e recuperar o seu Walkman da sala de professores, onde tinha sido guardado depois de confiscado. Nesta aventura Gonch é acompanhado pelo seu amigo «HOLLO» (PAUL HOLLOWAY) e tem um tempo limitado para efectuar a recuperação.

Este jogo necessita de bastante perícia e treino de modo a encontrar os vários objectos, usá-los no momento exacto e obter dos vários personagens as acções e informações correctas.

Um jogo que irá ser certamente um êxito.

CASCADE GAMES

IAN MARTIN o programador responsável por ACE desenvolveu para a CASCADE GAMES, SKY RUNNER.

O jogo desenrola-se no espaço interplanetário num tempo futuro em que a Terra foi reduzida à obediência sob o efeito de drogas. A missão de SKY RUNNER é libertar a humanidade acabando com o comércio de droga.

Até agora só temos notícia deste jogo nas versões para os Spectrum 48K e Commodore 64.

1 — COMMANDO
2 — ACE
3 — JACK THE NIPPER
4 — KUNG FU MASTER
5 — MATCH DAY II
6 — GREEN BERET
7 — PSY CHESS
8 — SPITFIRE 40
9 — BOMB JACK
10 — DAN DARE
11 — MOLECULA MAN
12 — HYPER SPORTS
13 — BATMAN
14 — RALLY DRIVER
15 — ROCKY
16 — POLE POSITION
17 — ROBIN HOOD
18 — MANIC MINER
19 — SIR FRED
20 — ZYTHUM
21 — CHUKIE EGG
22 — DYNAMITE DAN II
23 — ZORRO
24 — HEAVY MAGIC
25 — HAPPIEST DAYS
26 — GLADIATOR
27 — POPEYE
28 — PING PONG
29 — QUAZATRON
30 — KNIGHT TIME

CORREIO

CELSO BAPTISTA MARTINHO ANADIA

«Estimada revista SOFTFILE:

Sou assinante da vossa revista e desde já tenho a acrescentar que aprecio muito a organização e configuração da vossa revista. Finalmente temos uma revista portuguesa a um nível bastante aceitável.

O programa que vos envio, trata-se de um pequeno processador de textos da minha autoria e que costumo utilizar com os meus textos. Após ter redigido o texto (opção 1) terá que o armazenar numa das três memórias do processador (opção 2). A partir daí poderá optar por qualquer das outras opções. Se desejar corrigir algum texto apenas se terá de manifestar na opção.

A propósito, gostaria que me informassem se possível, se existe alguma cartridge ou cassette que transformasse o Commodore 64 num Spectrum 48K, e que aceitasse o seu software.

Acho que a vossa revista se inclina mais sobre os computadores Sinclair do que quaisquer outros. Acho que os leitores ficariam contentes se falassem mais de outros computadores como, por exemplo: COMMODORE, APPLE, MSX, AMSTRAD, etc.

Colaboro ainda com alguns Pokes para o COMMODORE 64-128 (M64).»

Amigo Celso antes do mais obrigado pela tua colaboração.

O teu processador de texto, por falta de espaço, não nos é possível incluir neste número da Revista Softfile. Está já programado para o próximo número desta.

Sinceramente não conhecemos nenhuma cartridge ou cassette que transforme o Commodore 64 num Spectrum 48K.

Sobre a revista Softfile não incluir artigos sobre outros computadores como o Commodore, Apple, MSX, Amstrad, é pela simples razão de não ser esta a sua ideia original. Estes computadores irão ser abordados na nossa outra revista a «SOFTFILE-PROFISSIONAL».

PRINT PEEK(O) — tecla premida no datassete
OPEN9,4:CMD9:LIST — Listagem de um programa para a impressora.
PEEK(1024 + col + 40*lin) — indica o caractere desta memória.

VIC-20:
Programa para ampliar o ecrã do VIC-20 (23X22) para (29X27):

```
POKE 43,15: POKE44,19: POKE 641,15: POKE642,19: NEW
1 N=36864: POKE N,7: POKE N+1,29: POKE N+2,27: POKE N+3,58
2 FOR T=0 TO 782: POKE 4096+T,32: POKE 37888+T,1: NEXT: NEW
```

COMMODORE-16
SYS50528 — Muda a cor da borda
SYS50496 — Muda a cor dos caracteres
SYS62116 — Reset total
SYS50501 — Muda a cor do papel
SYS32793 — Reset parcial

ALEXANDRE RIBEIRO QUELUZ

«Exmos. Senhores

ASSUNTO: Artigo técnico sobre BETA BASIC 3.0

Constituiu para mim uma agradável surpresa a leitura do número 2 da revista SOFTFILE pela qualidade dos artigos publicados e a sua apresentação gráfica. Considero bastante acertada a escolha dos sistemas de microcomputadores aos quais dão apoio na vossa revista, porque constituem a maioria dos microdomésticos em Portugal. As minhas felicitações também pelo facto de terem sabido resistir à via fácil e tentadora de serem mais uma revista dedicada à divulgação de jogos de computador para o SPECTRUM e similares.

Tomei a liberdade de submeter à vossa apreciação um artigo de apresentação do utilitário para o SPECTRUM, o BETA BASIC 3.0, que segue em anexo.»

Excelente o trabalho que nos enviou e que temos o prazer de publicar.

Constitui este um acto que os nossos leitores deverão seguir.

Aguardamos com ansiedade futuros trabalhos.

GERARDO LISBOA

«Caros senhores da revista 'Softfile'

É com muito prazer que sou assinante desta vossa revista que considero uma das melhores a nível nacional. Infelizmente verifico que lhe falta uma certa concepção gráfica mais apurada, que o aspecto das vossas páginas é um bocado cru, o que é bastante negativo para a vossa revista. Espero no entanto que consigam um maior esforço da parte da vossa equipa artística no sentido de melhorarem sempre.

Há também alguns erros de impressão cuja repetição impede por vezes a correcta leitura dos artigos. Por exemplo, no n.º 3, pagina 40, coluna 2, da linha 8 para a 9 o texto está cortado; importavam-se de me dizer o pedaço que falta?

Outro pormenor que me incomoda um bocado na vossa revista é o da numeração das páginas do "ESPAÇO 2" não ser diferenciada. Não se poderia numerá-la separadamente de modo a fazer de facto um suplemento?»

Amigo Gerardo, concordamos contigo quanto ao aspecto gráfico. Tentamos melhorá-lo. Contudo, nem sempre é possível fazê-lo a nosso contento porquanto se traduz em custos um pouco elevados. Com o tempo e gradualmente iremos certamente corrigir este pormenor.

Obrigado pela chamada de aten-

CORREIO

ção ao pormenor da gralha tipográfica. Conforme teu pedido transcrevemos a seguir o parágrafo completo.

«Pág. 40 — Revista Softfile Nº 3

2.1 — Análise sintática.

A extensão do BASIC do Spectrum é baseada no prolongamento da análise sintática das linhas de programa levando em conta os novos comandos.

Todos os erros detectados pelo Spectrum dão origem a um salto para o endereço 0008h, onde tem início a rotina de processamento de erros. A passagem do programa por este local activa o mecanismo de paginação e o programa de extensão fica em controlo do computador.

Quando é detectado um erro, numa linha de programa ou comando directo, o programa de extensão verfica se o «erro» foi provocado por um dos novos comandos, analisando a linha em causa a partir do local onde o Spectrum detectou o erro. Caso esta análise falhe, o estado do computador é reposto e o comando devolvido ao Spectrum, exactamente na mesma posição onde tinha sido retirado, prosseguindo o processamento normal de erro.

A sintaxe do novo comando é verificada e, quando o Spectrum se encontra em modo de execução, o comando é entregue à rotina respectiva. Quando esta termina é chamada uma rotina de processamento de fim de comando que pode, em alguns casos, entregar a iniciativa ao TOS (Ex: escrita no ecrã no caso de CAT*) e que termina processando os eventuais erros e devolvendo o comando ao Spectrum.

Caso o Spectrum esteja apenas em modo de análise sintática, por exemplo quando se está a criar um programa, o sinal de erro é eliminado e o controlo devolvido ao Spectrum que aceita a linha incluindo o novo comando.

Quando a análise da nova sintaxe falha, o controlo é devolvido ao Spectrum, por intermédio de um salto para o endereço guardado no variável de sistema VECTOR. Após um reset esta variável é inicializada com o endereço da rotina de retorno ao Spectrum, mas pode ser alterada para qualquer outro valor, dando a possibilidade ao utilizador de criar os seus próprios comandos com a sintaxe que desejar.

No fim dos novos comandos, o retorno ao Spectrum é feito por um salto para o endereço que se encontra na variável ABORT, o que permite ao utilizador interceptar o retorno ao Spectrum e alterar o modo como os comandos terminam. Esta facilidade é útil para alterar o processamento dos erros.

No capítulo de exemplos são desenvolvidas aplicações destas possibilidades.»

Sobre a organização da rubrica ESPAÇO 2 em suplemento vamos tentar introduzir a tua ideia nos nossos próximos números.

(Já agora um parêntesis acerca do ESPAÇO 2: Julgo ser a tradução dos manuais uma excelente iniciativa da vossa parte. Por um lado, porque o facto de os manuais virem normalmente em inglês dificulta o utilizador inexperiente. Por outro, dado que em muitas lojas não há manuais, o utilizador está normalmente 'às aranhas'. Mas este último facto só acontece porque na maioria das lojas só se vende cópias 'piratas'; será que não é viável um mercado legal de 'software' em Portugal?)

Penso que se estes problemas fossem resolvidos esta revista competiria muito facilmente com as congéneres estrangeiras. Já agora fico à espera da nova revista 'SOFTFILE-PROFISSIONAL'.

Como utilizador, desde cerca de há ano e meio, do sistema FDD da TIMEX e conhecedor do código máquina, fico-lhes desde já muito agradecido com a publicação do "MANUAL TÉCNICO" que me vem dar uma grande ajuda na programação directa do dito equipamento. No entanto pedia-lhes, caso fosse possível, que publicassem uma tabela com os códigos necessários para efectuar as diversas operações e outra tabela com os 'ports' utilizados no SPECTRUM e o significado de cada um dos seus 'bits' (não foram muito claros no último número). Esta última tabela tem uma razão de ser: é que as FDD, apesar de serem 'invisíveis' em questões de memória, não o são em questões de 'ports'. É por isto que os seguintes programas funcionam com dificuldade ou não funcionam mesmo, porque a busca ao teclado que eles realizam vai ler de alguma maneira os 'ports' das FDD, falseando os resultados. Os programas (que eu encontrei) são: JET SET WILLY II; THE ARTIST; THE RATS. Um exemplo das anomalias é a verificada no JSWII em que o boneco está sempre a saltar (como se estivessemos a carregar ininterruptamente na tecla de salto).

A «Softfile-Profissional» irá aparecer nas bancas uma semana depois da Softfile normal sair a público. Esperamos que gostes. Ainda não é o definitivo, queremos que seja de alto nível. Vamos tentar.

Quanto ao teu último ponto iremos fazer artigo sobre ele proximamente.

Obrigado.

Escreve!

MÁRIO CORTÊS QUATORZE COIMBRA

«Olá, chamo-me Mário João Cortês Quatorze e estou a escrever-vos para enviar um pequeno trabalho sobre o PROFANATION do qual publicaram um poster.

Queria antes agradecer-vos pela iniciativa que tomaram de elaborarem uma revista de tão elevado nível. Penso que deveriam dedicar-se um pouco mais ao comentário dos jogos (nunca se esqueçam de referir as teclas para cada um). Acabei o jogo MIKIE graças ao vosso

CORREIO

esclarecimento do que era necessário fazer na última revista.

Talvez pudessem abrir um bocado mais os conhecimentos que possuem (intermédio de pokes, rotinas, ensinar a trabalhar com compiladores, assembladores, desassembladores, etc...)

Em relação ao Profanation que considero um jogo muito bom:

Para ir de uma sala para outra:

— Teclar simultaneamente VICTOR, o jogo pára, introduzem o n.º de sala a que querem ir ter (de 1 a 45), depois de teclarem o n.º da sala que querem ir, fazer Enter, depois pomos o código 9127, e, Enter. Depois é só colocar o cursor que aparece, no sítio onde queremos, através das teclas O,P,Q,A, e no fim Enter.

O código 9127 é necessário só em algumas versões que sem este farão o ZX desligar.

Aqui envio uma rotina que permite jogar com vidas infinitas, ou 255 vidas, salvar quadros e tirar obstáculos.

Junto envio um quadro com um mapa e os n.ºs de cada sala.

Para salvar quadros (pantallas) teclar A e S.

Aguardo ansioso pela publicação deste trabalho caso ele seja aceite como tendo algum interesse.

Queria que publicassem um pequeno texto no Correio Club Soft:

Agradecia que o JORGE FONSECA, Mortágua, me contactasse, ele e todos os que tiverem histórias ou jogos em C/M que considerem de interesse; prémios superiores a 200 000$00. Através de conhecida firma de Software inglesa. Escrever para Mário João Cortês Quatorze — Apartado 364-3007, Coimbra Codex.

É tudo, continuem o excelente trabalho que estão a realizar.

Até breve.

Como se vê, as opiniões divergem. Uns pretendem que nos dediquemos mais aos jogos, outros à parte técnica. Vamos tentar contentá-los a todos.

Obrigado pelo teu trabalho. Cá esperamos o próximo.

MIGUEL ALVES LISBOA

«Aqui vos envio todas as minhas rotinas de jogos para o ZX Spectrum que possuo.

Todas as minhas rotinas ajudam a jogar os jogos respectivos, quase todas são de vidas infinitas ou coisa parecida.

Não entendo se os senhores pagam pela qualidade dos programas ou pelo seu comprimento; nos últimos meses ando a trabalhar num copiador de programas em código máquina, vai-se chamar «MICROCOPI», e penso poder enviar-vos antes do fim do mês de Setembro.

Espero que apreciem as minhas rotinas tanto como eu gosto da vossa revista.»

Em primeiro lugar queira aceitar as nossas desculpas por só agora lhe respondermos.

É espírito da equipa Softfile divulgar e apoiar todos os trabalhos dos seus leitores. Sobre a questão de «pagarmos pela qualidade dos programas ou pelo seu comprimento» orientamo-nos pelo espaço físico que ocupam em cada número da revista mas essencialmente pelo seu conteúdo/qualidade.

Esperamos com ansiedade o teu programa «MICROCOPI».

PEDRO PESTÂNÃ PÕRTO DE MÓS

«Caros Amigos,

Gostei muito como estava apresentada a vossa revista, com muita qualidade técnica, etc., mas vê-se aí uma coisa que poderia ser mudada, ou seja, os jogos que aí apresentam por vezes já são muito antigos, como por exemplo: Winter Games 1/2, mais um ou dois. Vou mandar uma lista de programas (jogos e não jogos). São já de há poucos meses. 1.º, Super Mais; 2.º, Winter Sport's; 3.º, I.C.U.P.S.; 4.º, Tenis; 5.º, Psy Chess; 6.º, Paper Boy; 7.º, Stainless Steel; 8.º, Kung Fu Master; 9.º, Graphic Adventure Creator; 10.º, TT Racer; 11.º, Knight Time; 12.º, Mindstone; 13.º, Colossus 4 Chess; 14.º, Tujad; 15.º, Johnny Reb II; 16.º, Pub Games; 17.º, Zythum; 18.º, Glider Rider; 20.º, Street Hawk; 21.º, Headcoach; 22.º, Revolution; 23.º, Trapdoor; 24.º, Great Escape; 25.º, Strike Force Cobra; 26.º, Danger Mouse in Hakin' Whooper; 27.º, Vera Cruz; 28.º, Trivial Pursuit; 29.º, Gestão de Stocks; 30.º, Dynamite Dan II. Espero que estes programas irão vencer aí. Porque são as últimas das últimas. Vou ainda mandar uma grande série de Vidas Infinitas. Em programas é mais simples: SAI COMBAT

```
POKE 65364,201 —
Pyracurse (Vidas Infinitas)
10 CLEAR 24799
20 PRINT    1:AT0,6: «Pôr gravador em andamento»
30 LOAD " " CODE
40 POKE 23325,201: RANDOMIZE USR 23299
50 POKE 33446201: RANDOMIZE USR 29600
FIM
Stainless Steel - POKE 46991,0
NIGHT MARE Rally: 100 MERGE " ":CLEAR 24791:
POKE 23785,201: RANDOMIZE USR 23760:
POKE 23785,221: POKE 26267,255: RANDOMIZE USR 23785

PAPER BOY
10 LOAD """" CODE
20 FOR F=65045 To 65054
30 READ A: POKE F,A: NEXT F
40 Randomize USR 65000
50 DATA 62, 100, 50, 143, 197, 62, 2, 50, 144, 197

THE PLANETS
 Fazer MERGE """" e colocar antes do randomize USR:
FOR F=24963 to 40011 STEP 342.
POKE F,0: NEXT F
```

Tudo bem. Tomámos nota da tua crítica.

Esperamos que continues a colaborar connosco.

Até à próxima.

ROTINA PARA ESCREVER NA PISTA 04 (DIRECTORIA DA DISKETTE)

Este pequeno programa em assembler destina-se, fundamentalmente, a programadores com experiência em código máquina e que possuam um sistema de discos FDD ligado ao seu microcomputador TIMEX.

O programa permite, facilmente, escrever directamente na pista da directoria iniciando-se com a rotina WTDIR que efectua a paginação da memória do interface através de CALL PAGEIN, com 0 no registo IY, correspondendo a um CALL ao endereço 0008h, indispensável para seleccionar a página de extensão da ROM do interface.

A instrução LD DE, BUFFER indica o local disponível em RAM.

A rotina LOOP é a base do programa e efectua a transferência de dados do seguinte modo: carrega no registo HL o endereço do buffer de dados e no registo BC o seu comprimento passando-os para o interface através da instrução LDIR e para o controlador com a instrução CALL PUTDAT.

Efectua-se a seguir um salto para a rotina WTSEC que transfere os dados para o disco colocando o código de comando de escrita no registo A e transferindo-o de seguida para o BUFCOM.

O comado é enviado com CALL PUTCOM e finalizado com CALL RESPOSTA sendo o código de erro devolvido ao endereço BUFCOM + 2.

O programa volta então à rotina LOOP analisando se houve erro ou acabou a escrita. Se sim, devolve o controlo ao computador, se não, incrementa o sector e reinicia o ciclo.

Eng. Carlos Oliveira

ROTINA

```
WTDIR  PUSH IY
       LD   IY,0
       CALL PAGEIN
       POP  IY
       LD   DE,BUFFER
       LD   B,0
 LOOP  PUSH BC
       LD   HL,BUFDAT
       LD   BC,0100
       EX   DE,HL
       LDIR
       EX   DE,HL
       LD   A,FF
       CALL PUTDAT
       POP  BC
       CALL WTSEC
       LD   A,(BUFCOM+2)
       AND  A
       JR   NZ,ERROR
       LD   A,B
       ADD  07
       AND  OF
       LD   B,A
       JR   NZ,LOOP
       XOR  A
       LD   B,0
       LD   C,A
ERROR  JP   PAGEOUT
WTSEC  PUSH DE
       PUSH BC
       LD   E,B
       LD   D,4
       LD   C,0
       LD   A,1C
       LD   6BUFCOM),A
       CALL PUTCOM
       CALL RESPOSTA
       POP  BC
       POP  DE
       RET
```

SOFTWARE

Totoloto

Totoloto é um programa feito por portugueses e editado agora na versão 2.1 pela SOFTEC.

O programa permite executar desdobramentos e efectuar a análise estatística dos resultados do totoloto.

Em termos de desdobramentos pode efectuar 21 tipos diferentes divididos em 4 sistemas qu dão variadas hipóteses de acerto nos vários prémios (garantia de acerto no 1.º, 3.º, 4.º ou 5.º prémio).

Os números a serem desdobrados podem ser gerados aleatoriamente pelo computador ou introduzidos pelo operador.

Os desdobramentos podem ser de 3 tipos distintos consoante a quantidade de números fixos e que podem ir de zero a dois. Os números fixos podem também ser introduzidos pelo operador ou gerados aleatoriamente pelo computador.

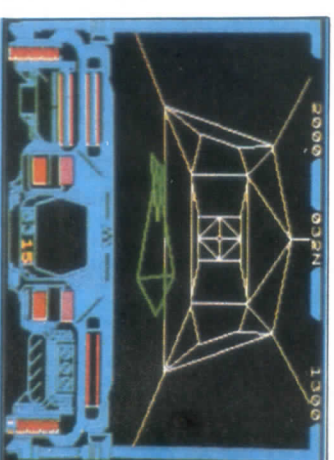
Os desdobramentos podem ser guardados em cassette para posterior consulta.

Os dados estatísticos podem também ser gravados e lidos em cassette. Conjuntamente com estes dados pode ainda optar pela introdução de informações complementares sobre o concurso, tais como o número do concurso; data e número de prémios saído.

Os vários valores podem ser imprimidos, dada a compatibilidade do programa com as impressoras TS 2040 e EPSON GP-50S.

Comentário — A comprar só se interessar pelo totoloto.

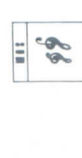
SOFTWARE

Starglider

Trata-se de um jogo de acção no espaço, programado por alguns maiores especialistas deste tipo de jogos a três dimensões e responsáveis por sucessos como «STAR STRIKE» e «STAR TRIKE II».

O resultado é deveras espantoso as naves inimigas que aparecem no ecrã parecem tiradas do filme Guerra das Estrelas, constituem o exemplo mais perfeito da utilização dos chamados vectores gráficos animados, imprimindo aos vários objectos uma perfeita sensação de três dimensões. De salientar ainda o notável efeito de profundidade dado pelo painel de instrumentos, e a extraordinária sequência de lançamento de um mísil.

Mas vamos à história. A acção desenrola-se no planeta Novenia, provido de um sistema de defesa completamente inexpugnável e preparado para repelir qualquer objecto que se aproxime do planeta. Com uma única excepção, as defesas estavam programadas para deixarem entrar «STARGLIDER», um pássaro raro que escolhia Novenia como ponto de paragem na sua rota migratória. Aproveitando este facto, os Egrans disfarçaram as suas naves de batalha, com a forma do pássaro «STARGLIDER», conseguindo passar as intransferíveis defesas de Novenia. São estas as naves dos Egrans que Jasen e Katia a bordo de uma nave vão enfrentar.

Um pormenor curioso este jogo é vendido no Reino Unido em manuais bastante completos, e um livro de 64 páginas contando a história de Novenia.

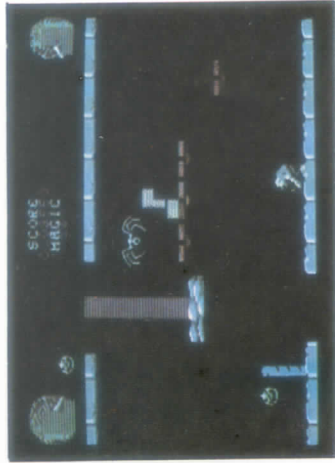
Análise geral: Um excelente jogo de compra obrigatória.

POKES

NOTAS

POKES

NOTAS

UNI MICRO
microcomputadores
software
vídeos
Rua da Misericórdia, 67-1.º/Dto.
Telfs. 327073/362431
1200 Lisboa

SOFTWARE

Cauldron II

CAULDRON II é o tipo de jogo que também se poderia chamar «E A ABÓBORA CONTRA-ATACA» — se na primeira versão deste jogo fazíamos o papel de uma feiticeira que massacrava umas simpáticas abóboras sobreviventes, que desejavam ardentemente vingar-se da dita bruxa. E para concretizar esta terrível ameaça, a abóbora pretende encontrar um punhado de cabelos de bruxa (!), para os lançar no caldeirão da poção mágica... Atroz, sem dúvida: onde é que já se viu uma poção mágica cheia de cabelos?!

A acção decorre no interior de um enorme castelo, em que cada sala, normalmente infestada de diversas criaturas, constitui um novo obstáculo ao avanço da nossa abóbora.

Este jogo peca sobretudo por uma certa falta de imaginação, e tanto os gráficos como o som são perfeitamente medianos.

Por último, resta apenas referir que a editora deste programa é a PALACE SOFTWARE, que além da versão para o Spectrum 48K editou ainda uma segunda para o Commodore 64.

Análise geral: Jogo bastante fraco. A evitar.

computadores	Sinclair - TC 2048 - TC 2068 - Commodore 64
joystick	(crossed out)
software	Palace Software
acção	
jogadores	1
grafismo	m ~~r~~ b mb
som	(circled)
comandos	a definir pelo utilizador
apreciação	m (r) b mb

SOFTWARE

Druid

Do alto da sua sinistra torre negra a princesa das trevas exerce a sua influência malévola sobre todo o seu vasto domínio com a ajuda de quatro mágicos diabólicos.

Mas um druido vai tentar derrotar as forças maléficas e penetrar na torre.

Armado com os feitiços da água, do fogo e da electricidade o druido tenta derrotar os fantasmas e gigantes que povoam os bosques ao redor da torre, antes que o seu poder enfraqueça demasiado e a sua fraqueza se torne fatal deve encontrar os arcos mágicos e retirar só um feitiço, é necessária muita atenção para escolher o feitiço ou o objecto adequado à situação.

Os feitiços que pode escolher são: a invisibilidade, de duração limitada mas que lhe permite passar sem perigo pelos inimigos. O encantamento, talvez o feitiço mais poderoso, que lhe aumenta o poder mágico, ao mesmo tempo que destrói todas as criaturas diabólicas presentes no écran. E o feitiço do «golem», que chama esta criatura poderosa para defender o druido, com o seu corpo dos ataques inimigos. Além destes feitiços o druido pode recuperar as suas forças repousando sobre misteriosos pentagramas espalhados pela floresta.

Ainda que possuido de numerosos poderes e feitiços a missão do druido poderá parecer fácil, nada mais errado, os inimigos são numerosos e poderosos, a floresta e a torre autênticos labirintos que é necessário vencer.

O jogo é realmente extraordinário, possuidos de excelentes gráficos, e deveras excitante de jogar.

Comentário — A comprar! Obrigatoriamente.

computadores	TC 2048 - TC 2068 - SINCLAIR
joystick	Kempston
software	Firebird
acção	
jogadores	1
grafismo	m r (b) mb
som	
comandos	a definir pelo utilizador
apreciação	m r b (mb)

POKES

NOTAS

POKES

NOTAS

microcomputadores
software
vídeos

Rua da Misericórdia, 67-1.º/Dto.
Telfs. 327073/362431
1200 Lisboa

Todas as tardes

SÉRIE) DIÁRIO — N.º 5843 — 1986 — QUINTA-FEIRA

A CAPITA

Director: RODOLFO IRIARTE

ESCARAFU

VENHA

A

BADA

ENTE

ODA

CONV

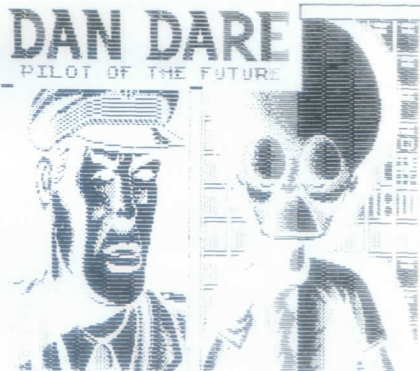
PO

CA

o seu JORNAL

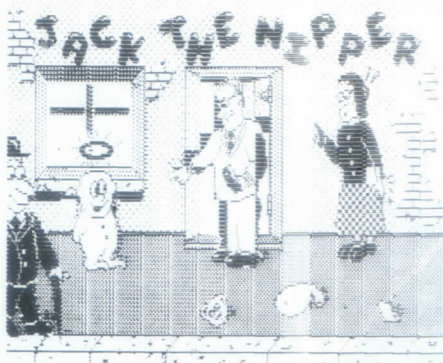
POKES

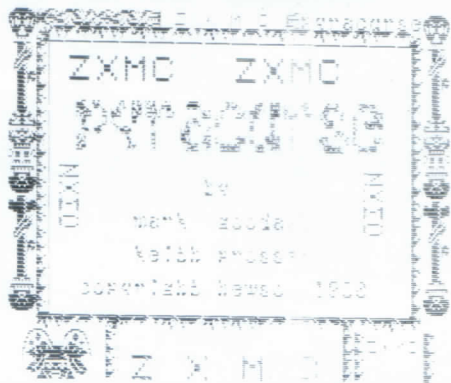
ALERTA A TODOS OS «POKISTAS»!

Esta rubrica está aberta a todos aqueles que queiram enviar os seus pokes ou dicas sobre os «pequenos segredos» que possibilitam a cada um «chegar» ao final de cada jogo (como passar de um ecrã, de um nível para outro, etc...)
Escreve para a Revista Softfile — Secção Pokes
R. Rodrigo da Fonseca, 95-4º. — 1200 LISBOA.
CONTAMOS CONTIGO.

DAN DARE

```
10 BORDER 0:INK 0:PAPER 0:
  CLS: CLEAR 25599
20 RESTORE
30 FOR i=23296 TO 23311
40 READ a: POKE i,a
50 NEXT i
60 PRINT AT 0,6;
  "TECLE PLAY"
70 RANDOMIZE USR 23296
80 POKE 65326,201
90 RANDOMIZE USR 65032
100 POKE 47710,201
110 POKE 46885,201
120 POKE 43526,0
130 POKE 42863,0
140 POKE 42111,0
150 RANDOMIZE USR 39000
500 DATA 221,33,171,253,17,9,3
510 DATA 62,255,55,205,86,5
520 DATA 48,241,201
```

JACK THE NIPPER

```
10 RESTORE
40 LET w=1
50 LET tot=0
60 FOR i=49998 TO 50070
70 READ a: LET tot=tot a * w
80 LET w=w+1
90 POKE i, a
100 NEXT i
110 IF tot <> 319540 THEN
PRINT «ERRO NA DATA»:
BEEP 1,1:STOP
120 PRINT = 1;AT 14,1;
  "TECLE PLAY"
130 RANDOMIZE USR 49998
1000 DATA 0,221,33,203,92
1010 DATA 17,116,1,62,255
1020 DATA 55,205,86,5,210
1030 DATA 78,195,33,135,195
1040 DATA 17,213,93,1,17
1050 DATA 0,237,176,33,35
1060 DATA 191,237,99,26,93
1070 DATA 33,209,255,237,99
1080 DATA 33,93,175,61,50
1090 DATA 167,93,33,195,209
1100 DATA 237,99,165,93,195
1110 DATA 0,93,175,50,0
1120 DATA 170,33,0,0,237
1130 DATA 99,1,170,175,211
1140 DATA 254,55,201
```

SGRIZAM
(vidas infinitas)

```
POKE 34548,0
POKE 29534,0
POKE 29519,0
```

PYRACURSE

```
10 CLEAR 24799: LOAD "" 
  CODE 23296
30 POKE 23325,201: RANDOMIZE
  USR 23299
40 POKE 33450,201: RANDO-
MIZE
  USR 29600
```

AUTOR: ENRIQUE CERVERA

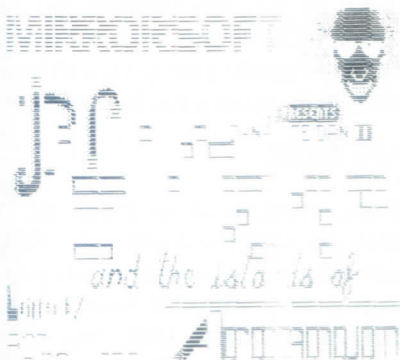
COPYRIGHT DINAMIC 1986

PHANTOMAS II

```
POKE 26606,0
POKE 28452,0
POKE 29756,0
POKE 31004,1
```

POKES

KNIGHT TYME

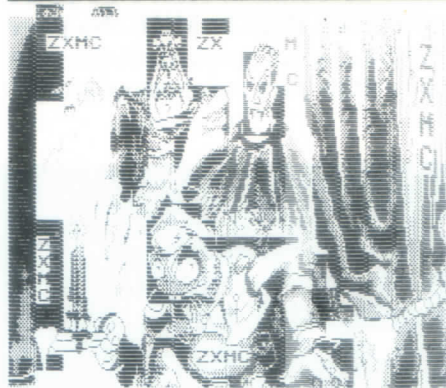
```
POKE 24584,255
POKE 24585,255
POKE 45322,255
POKE 45323,255
POKE 41456,0
POKE 41457,0
```

DYNAMITE DAN II

```
10 FOR F = 60000 TO 60049
11 READ A: POKE F,A
12 NEXT F
15 PRINT AT 10,2;
 "TECLE PLAY"
16 RANDOMIZE USR 60000
20 DATA 49,23,94,221,33
21 DATA 64,156,17,144,1
22 DATA 175,55,205,86,5
23 DATA 48,242,33,132,234
24 DATA 17,56,91,1,20
25 DATA 0,237,176,62,52
26 DATA 50,71,156,195,64,156
28 DATA 33,0,24,34,74,113
30 DATA 62,201,50,104,115
32 DATA 195,144,108,0,0
33 DATA 0,0,0
```

PAPER BOY VIDAS INFINITAS

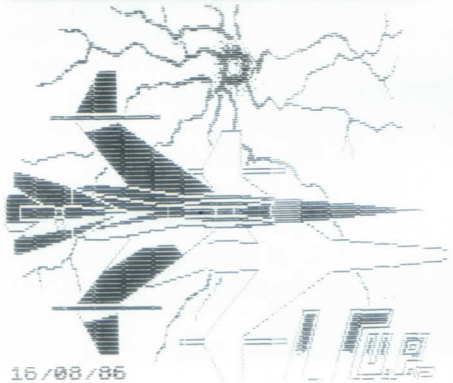
```
10 LOAD "" CODE
20 FOR f = 65302 TO 65307
30 READ a: POKE f,a: NEXT f
40 POKE 65110,22: POKE 65111,
 255
50 RANDOMIZE USR 65000
60 DATA 50,151,187,195,86,5
```

NEXOR

```
10 CLEAR 65535: FOR u = 23296
TO 23317: READ a: POKE u,a:
 NEXT u
30 RANDOMIZE USR 23296
40 FOR u = 63916 TO 63922:
 READ a: POKE u,a: NEXT u
50 RANDOMIZE USR 63744
60 DATA 62,255,55,221,33,229,
 248,17,137,1,205,86,5
70 DATA
48,243,33,172,249,34,112,
 249,201
80 DATA 175,50,116,141,195,0,120
```

ACTION REFLEX

```
5 CLEAR 27000
10 PRINT = 1: AT 0,4;
 "TECLE PLAY"
20 MERGE ""
30 POKE 23988,54:POKE
23349,201
```

I.C.U.P.S.

```
1 CLEAR 65535
2 PRINT AT 21,6; "TECLE
 PLAY"
3 MERGE ""
4 RUN 5
15 FOR x = 0 To 26
45 DATA 221,33,0,91,17,0,117,62
46 DATA 255,183,55,8,243,205,
 98,5
47 DATA 210,0,0,62,200,50,245,
 134
48 DATA 195,185,134
```

PHANTOMAS I

```
POKE 44819,0
POKE 46790,191
POKE 52290,0
POKE 45126,195
POKE 45127,16
POKE 45128,175
POKE 48370,0
POKE 46720,58
POKE 46723,58
POKE 45996,0
```

Micro-análise

ZX SPECTRUM 128K

De aspecto semelhante ao ZX Spectrum+, utilizando nomeadamente o mesmo tipo de teclado, o Spectrum 128K, inicialmente fabricado em Espanha pela Investronica, constitui a evolução lógica dos microcomputadores da linha Sinclair, utilizando microprocessadores Z80 de 8 bits, aumentando a sua capacidade de memória, introduzindo uma série de funções extra e mantendo simultaneamente a total compatibilidade de modo a poder utilizar toda a vasta gama de software existente para o ZX Spectrum 48K.

HARDWARE

Num primeiro contacto o Spectrum 128K apresenta-se com um aspecto exterior semelhante ao do já bem conhecido ZC Spectrum+, no entanto uma visão mais cuidada revela-nos certas diferenças, as mais óbvias são o logotipo 128K, colocado no canto direito e que constitui quanto a nós, um pormenor negativo neste computador, pois se por um lado permite uma melhor dissipação de calor, evitando avarias por sobreaquecimento, por outro após algum tempo de funcionamento, o aquecimento do dissipador torna-se excessivo e ao tocar-lhe pode tornar-se doloroso. Julgamos que com um pouco de imaginação, os engenheiros da Sinclair poderiam encontrar uma melhor solução.

Além do teclado principal existe ainda um pequeno teclado numérico (keypad) que pode ser ligado ao painel frontal do computador e que faculta funções de cálculo e edição. No aspecto exterior várias coisas mudaram: Em relação ao Spectrum+, os portos de MIC e EAR aparecem agora do lado esquerdo junto do botão de reset e do porto combinado RS232 e MIDI. Na parte traseira aparece agora, também, uma saída RGB para monitor policromático.

No seu interior, o 128K apresenta além do Hardware típico do ZX Spectrum 48K (incluindo um microprocessador Z80 funcionando a 3.5 MHZ) uma série de hardware adicional, como sejam a memória que passou a 128K de RAM, divididos em páginas de 16K bytes e 32K bytes ROM, 16K Bytes para o modo 48K e 16KBytes para o novo modo 128K. O som apresenta-se extraordinariamente melhorado, devido a utilização de um novo processador de som, o AY-3-8912, funcionando com três canais separados passando o som a ser reproduzido através da TV, de uma alta fidelidade ou de um simples gravador ligado à saída MIC tendo desaparecido definitivamente o altifalante do interior do computador. Foi criado um novo comando de som PLAY, em substituição do anterior BEEP. Este novo comando deve ser seguido por uma «string» contendo a programação musical.

RAM DISK

Para ultrapassar o facto de o espaço de endereçamento do Z80, assim como de qualquer microprocessador de 8 bits, ser somente de 64KB, o Spectrum 128K usa uma técnica designada «bankswitched memory system», em que a memória é dividida em páginas de 16KB, sendo qualquer posição no espaço de

(continua na pág. 35)

MANUAL TÉCNICO DAS FLOPPY DISK DRIVE TIMEX

CONT. NÚMERO ANTERIOR

POR cortesia da TMX PORTUGAL LTD., continuamos neste número a publicação do manual técnico do sistema FDD (FLOPPY DISK DRIVE) TIMEX.

É nossa intenção ao publicá-lo, ajudar a maioria dos utilizadores deste sistema a melhor compreendê-lo e utilizá-lo de modo a conseguir explorar as suas inúmeras capacidades e possibilidades.

3.b — SAVE e LOAD

Estas funções talvez sejam as que têm interesse mais imediato para o programador. São executadas pelas rotinas SAVEP e LOADP e dado o número de parâmetros necessários para as definir, são consideravelmente mais difíceis de usar que as anteriores.

A rotina SAVEP requer os parâmetros seguintes:

A	— tipo de save 0: programa. 1: array numérico. 2: array de caracteres. 3: código.
BC	— número de bytes a gravar.
DE	— endereço inicial.
HL	— número da linha no caso de programa c/ LINE n.
A'	— comprimento do nome de BUFDAT (incluído o O final).
(BUFDAT)	— nome do ficheiro.

A rotina que se segue, grava no disco, na directoria corrente, o conteúdo do écran sob o nome de EXEMPLO.SCR.

```
;
;TIMEX-SOFT
;
;ROTINA DE SAVE DE ECRAN
;
;Retorna com o código de erro no acumulador (0 se tudo bem)
;
;
;
BUFDAT     EQU   2000H
BUFCOM     EQU   2100H
;
PAGEIN     EQU   0008H
PAGEOUT    EQU   0603H
SAVEP      EQU   0620H
;
SCREEN     EQU   16384            ;endereço do écran
SCRLEN     EQU   6912             ;comprimento em bytes do écran
;
;
SAVESCR    PUSH  IY               ;paginação
           LD    IY,0
           CALL  PAGEIN
           POP   IY
;
           LD    HL,TEXTO         ;ponteiro para o nome do ficheiro
           LD    DE,BUFDAT        ;buffer de comunicações
           LD    BC,COMP          ;comprimento do nome
           LDIR                   ;transferência do nome
           EX    DE,HL
           LD    (HL),0           ;marca de fim de nome
;
           LD    A,COMP+1         ;comprimento do bloco a transf.
           EX    AF,AF            ;primeiro parâmetro
           LD    HL,0             ;não e usado
           LD    DE,SCREEN
           LD    BC,SCRLEN
           LD    A,3              ;tipo código
           CALL  SAVEP            ;executa o comando
           LD    A,(BUFCOM+2)     ;código de erro
           LD    B,0
           LD    C,A              ;passagem do erro para o BASIC
           JP    PAGEOUT          ;retorno e paginação
;
TEXTO      DEFM  "EXEMPLO.SCR"    ;nome do ficheiro
;
COMP       EQU   $-TEXTO
;
           END
```

Para segundo exemplo, vamos executar a função inversa usando a rotina LOADP. Esta rotina requer parâmetros diferentes:

(HEADER) — tipo de ficheiro pretendido.
(HEADER + 1) — número máximo de byte a carregar (se 0 e usado o comprimento total do ficheiro).
(HEADER + 3) — endereço alternativo de carga (se 0 e usado o endereço que consta do ficheiro).
(BUFDAT) — nome do ficheiro.
B — comprimento do nome incluindo o 0 final.

TOP SECRET

```
;
;TIMEX-SOFT
;
;ROTINA DE LOAD DE ECRAN
;
;Retorna com o código de erro no acumulador (0 se tudo bem)
;
;
BUFDAT    EQU   2000H
BUFCOM    EQU   2100H
HEADER    EQU   214DH
;
LOADP     EQU   0623H
PAGEIN    EQU   0008H
PAGEOUT   EQU   0603H
;
;
LOADSCR   PUSH  IY                 ;paginação
          LD    IY,0
          CALL  PAGEIN
          POP   IY
;
          LD    HL,TEXTO           ;nome do ficheiro
          LD    DE,BUFDAT          ;buffer de comunicações
          LD    BC,COMP            ;comprimento do nome
          LDIR
          EX    DE,HL
          LD    (HL),0             ;marca de fim de nome
;
          LD    B,COMP+1           ;comprimento incluindo o 0
          LD    HL,0
          LD    (HEADER+1),HL      ;usar o valor de ficheiro
          LD    (HEADER+3),HL      ;idem
          LD    A,3                ;tipo código
          LD    (HEADER),A
          CALL  LOADP              ;executa o comando
          LD    A,(BUFCOM+2)       ;código de erro
          LD    B,0
          LD    C,A                ;passagem do erro para o BASIC
          JP    PAGEOUT            ;retorno e paginação
;
TEXTO     DEFM  "EXEMPLO.SCR"      ;nome do ficheiro
;
COMP      EQU   $-TEXTO
;
          END
```

No caso de LOAD de um programa a ROM de extensão encarrega-se de eliminar o programa anterior e instalar o novo. Para se executar um MERGE o processo é ligeiramente mais complexo. A rotina seguinte faz o MERGE do ficheiro PROG.BAS.

```
;
;TIMEX-SOFT
;
;ROTINA DE MERGE DE PROGRAMAS
;
;Retorna com o código de erro no acumulador (0 se tudo bem)
;
BUFDAT    EQU   2000H
BUFCOM    EQU   2100H
CHAIN     EQU   2138H
HL_TMP    EQU   213FH
HEADER    EQU   214DH
;
LOADP     EQU   0623H
PAGEIN    EQU   0008H
PAGEOUT   EQU   0603H
;
;
MERGE     PUSH  IY              ;paginação
          LD    IY,0
          CALL  PAGEIN
          POP   IY
;
          LD    HL,TEXTO        ;nome do ficheiro
          LD    DE,BUFDAT       ;buffer de comunicações
          LD    BC,COMP         ;comprimento do nome
          LDIR
          EX    DE,HL
          LD    (HL),0          ;marca de fim de nome
;
          LD    B,COMP+1        ;comprimento incluíndo o 0
          LD    A,80H
          LD    (CHAIN),A       ;sinal de MERGE
          LD    A,0             ;tipo programa
          LD    (HEADER),A      ;os outros parâmetros normalmente
                                ;usados não são necessários
          CALL  LOADP           ;executa o comando
          LD    A,(BUFCOM+2)    ;código de erro
          AND   A
          LD    B,0
          LD    C,A             ;passagem do erro para o BASIC
          JP    NZ,PAGEOUT      ;houve erro em LOADP
;
          LD    HL,(HL_TMP)     ;ponteiro p/ início do novo programa
          CALL  PAGEOUT         ;paginação
          CALL  08CEH           ;rotina de merge do Spectrum
          LD    BC,0            ;não houve erros
          RET                   ;fim da rotina
;
TEXTO     DEFM  "PROG.BAS"      ;nome do ficheiro
;
COMP      EQU   $-TEXTO
;
          END
```

3.c — Acesso directo ao conteúdo do disco.

No sistema FDD, a informação é gravada nos discos segundo o formato seguinte: 40 pistas, cada uma com 16 sectores, numerados de 0 a 15, cada um com 256 bytes. As primeiras 4 pistas, são reservadas para o Sistema Operativo TOS e a quinta para a directoria. Nas unidades de disco com 1 Mb, a serem lançadas em breve, a estrutura é idêntica excepto que, existem 160 pistas com as pistas pares no lado 1 e as ímpares no lado 2.

Apesar dos sectores estarem numerados sequencialmente de 0 a 15, a informação é guardada pelo TOS com um intervalo de 7 entre sectores consecutivos, de modo a melhorar o tempo de acesso. Por exemplo, um ficheiro que ocupe 512 bytes e cujo primeiro sector é o número 3, tem o segundo no sector 10.

ESPAÇO 2

BETA BASIC 3.0

Publicamos nesta edição um artigo enviado pelo nosso leitor Alexandre J. de Azevedo Ribeiro sobre o utilitário, para o ZX SPECTRUM, «BETA BASIC 3.0» pelo que publicamente o felicitamos pelo excelente trabalho e incitamos todos os outros a colaborarem também, se o pretenderem, nesta rubrica.

INTRODUÇÃO

O **BETA BASIC 3.0** da empresa inglesa BETASOFT, é um utilitário em código máquina de expansão do SINCLAIR BASIC do Spectrum, que lhe confere uma potencialidade de desenvolvimento de software em BASIC, em nada inferior ao que se pode encontrar em microcomputadores muito mais sofisticados. No entanto, contrariamente ao que sucede com a maior parte dos utilitários para o Spectrum, que são obstructivos ao desenvolvimento de programas logicamente encadeados e de fácil leitura (**RANDOMIZE USR xxx** é qualquer coisa que quebra uma sequência de programa), o **BETA BASIC 3.0** comporta-se como um novo «sistema operativo» bem mais potente que o actualmente existente no Spectrum.

Passamos a analisar em seguida os aspectos mais marcantes deste utilitário. Logo em seguida ao carregamento do **BETA BASIC**, quer de cassette (LOAD "") ou Microdrive (run) o Spectrum fica pronto a aceitar comandos. Aqui começam as surpresas, pois agora os comandos devem ser introduzidos letra por letra (que consegue ser mais rápido e menos sujeito a erros do que a utilização das keywords do **BASIC** normal) e não interessa se em maiúsculas ou minúsculas nas palavras que constituem keywords.

COMANDOS

O BETA BASIC 3.0 reconhece todos os comandos do **SINCLAIR BASIC** (os quais foram objecto de melhoramentos bastante significativos), bem como um conjunto de novos comandos que constituem a expansão da linguagem.

De um modo geral podem-se agrupar os comandos do **BETA BASIC 3.0** nas seguintes categorias:

ESPAÇO 2

— comandos de edição de programas
— comandos de armazenamento (cassette e Microdrive)
— comandos de tratamento e manipulação de dados
— comandos de gráficos
— comandos de utilitários diversos (Toolkits)

COMANDOS DE EDIÇÃO DE PROGRAMAS

Agora a edição de programas em **BASIC** deixou de ser aquela tarefa lenta e penosa de executar, com a utilização dos seguintes comandos:

— **AUTO — (AUTO m,n)** — permite a numeração automática de linhas de programa, a começar na linha m **e com incrementos de** n. **Deste modo o programador apenas tem que se preocupar com a escrita das instruções de cada linha de programa. A numeração automática termina quando o número de linha for inferior a 10 ou superior a 9983 e ainda quando o sistema imprime qualquer relatório.**

— **DELETE — (DELETE m TO n)** — apaga do programa actualmente em memória o conjunto de linhas de **m** a **n**, inclusive as linhas **m** e **n**.

EDIT — (EDIT m) — este comando copia a linha **m** para a zona de edição, independentemente da posição actual do cursor, contrariamente ao que sucede com o comando de **EDIT** do **SINCLAIR BASIC.** Após a edição, essa linha passa a ser linha actualmente acedida pelo cursor.

CURSOR — (movimentos do cursor) — durante a edição de uma linha de programa ou de comando directo, o cursor reage de acordo com o sentido das setas de movimentação de cursor, isto é, desloca-se ao longo de uma linha (até ao princípio ou fim da linha) e desloca-se também verticalmente, saltando de linha para linha. Esta liberdade de movimento do cursor facilita grandemente a correcção da linha actualmente a ser editado, ou o posicionamento do cursor numa determinada linha de programa.

LIST e LLIST — (LIST m TO n ou LLIST m TO n) — de modo semelhante ao comando **DELETE,** só se visualizam as linhas de programa compreendidas entre **m** e **n**. (**m** e **n** inclusive).

JOIN — (JOIN m) — concatena a linha m e a que lhe é imediatamente contígua e atribui-lhe o número de linha **m**. Caso não seja dado qualquer parâmetro **m**, o **JOIN** concatena a linha actualmente apontada pelo cursor com a que lhe é imediatamente contígua.

COMANDOS DE ARMAZENAMENTO

O BETA BASIC 3.0 não introduziu novos comandos de armazenamento, mas melhorou consideravelmente os já existentes. Esses melhoramentos incidiram principalmente na sintaxe dos comandos relacionados com os periféricos do **INTERFACE 1** (Microdrives, canal série e network).

Assim é possível (e desejável) definir um periférico principal **(default)**, com o qual se pretende dialogar mais frequentemente mediante o comando **DEFAULT**. Deste modo, se tencionamos trabalhar intensivamente com o Microdrive 1, podemos defini-lo como periférico principal por meio de **default = m1** e todos os comandos de acesso a ele podem ser dados por **LOAD/SAVE «filename»** em vez de **LOAD/SAVE *''; 1; «filename»**.

Os outros defaults admissíveis são:

— **default = t — cassette.**
— **default = b — canal série (RS 232).**
— **default = n — network.**

De notar que **m** e **n** devem ser seguidos de um número que indica qual o periférico seleccionado. Para **m** o número está compreendido entre 1 e 8, e para **n** entre 1 e 64.

O **BETA BASIC 3.0** permite fazer o **SAVE** e o **VERIFY**, quer de blocos de programa, quer da zona de variáveis independentemente do programa. A sintaxe dos comandos é a seguinte:

— **SAVE/VERIFY m TO n; «filename»** — faz o **SAVE/VERIFY** do conjunto de linhas de programa compreendidas entre o **m** e **n** para o ficheiro **filename** no periférico de serviço.

SAVE/VERIFY DATA; «filename» — faz o **SAVE/VERIFY** da zona de variáveis do programa

actualmente em memória para o ficheiro **filename** no periférico de serviço.

O comando de **LOAD** não tem qualquer melhoramento em relação ao actual, mas o comando **MERGE** ganha particular importância em função das opções de armazenamento que o **BETA BASIC 3.0** permite. Além disso, o comando **MERGE** funciona correctamente com os programas em modo *auto run* armazenados em Microdrive.

COMANDOS DE TRATAMENTO E MANIPULAÇÃO DE DADOS

Além da possibilidade de se poder armazenar a zona de variáveis independentemente do programa, o **BETA BASIC 3.0** suporta três tipos de acesso às variáveis de um programa, a saber:

— atribuição de valores por defeito sem utilizar a instrução **LET**.

— visualização das variáveis.

— manipulação de variáveis.

A atribuição de valores por defeito utiliza o comando **DEFAULT** em comandos do tipo **DEFAULT variável1 = valor1, variável2 = valor2,**, etc.

A visualização das variáveis é possível por meio dos seguintes comandos:

— **LIST DATA** — faz o display de todas as variáveis definidas.

— **LIST VAL** — faz o display de todas as variáveis numéricas definidas.

— **LIST VAL$** — faz o display de todas as variáveis do tipo «cadeia de caracteres» (strings) definidas.

Os comandos de manipulação de variáveis são os seguintes:

— **JOIN t$ TO s$** — permite inserir na string (ou array de caracteres) s$ a string (ou array de caracteres) t$. A inclusão é feita em qualquer posição da variável s$ (ela pode ser expandida, mesmo se for um array), e a variável t$ deixa de existir.

— **COPY t$ TO s$** — copia a variável t$ para a variável s$, de modo semelhante a **JOIN**, mas a variável t$ continua a existir.

— **EDIT variável** — permite editar a variável de uma maneira idêntica à edição de linhas de programa e comandos.

COMANDOS GRÁFICOS

É no conjunto de comandos gráficos que o **BETA BASIC 3.0** introduz uma nova dimensão à manipulação do ecrã do **SPECTRUM.** Os comandos de gráficos permitem:

— definir janelas de impressão (windows) tal como no **QL.**

— manipular a informação contida nas janelas, deslocando-a na horizontal (**horizontal scrolling**) e na vertical (**vertical scrolling**).

— imprimir em alta resolução mensagens em qualquer posição do ecrã.

— guardar a informação contida numa janela (em modo de alta resolução) para ser posteriormente impressa (em alta resolução) em qualquer posição do ecrã.

— **«pintar» (Fill)** rapidamente figuras fechadas de qualquer dimensão.

Todos estes comandos funcionam com a resolução de um pixel. Para além deles foram mantidos todos os comandos de baixa resolução (caractere) já existentes para o SPECTRUM, **e melhorados os de alta resolução, nomeadamente os comandos DRAW e PLOT.** O comando **PLOT** suporta o plot (impressão em alta resolução) de strings, e o comando **DRAW** permite traçar segmentos de recta com coordenadas relativas à última posição de **DRAW** ou **PLOT.**

COMANDOS DE UTILITÁRIOS

Na categoria de utilitários englobam-se todos aqueles comandos que facilitam o desenvolvimento, teste e correcção de programas, e os que, não sendo essenciais, proporcionam ao utilizador um ambiente de trabalho mais «confortável». De entre eles destacamos:

— **DEF KEY** — permite programar o teclado como teclas de funções, já que a cada tecla se pode atribuir uma sequência qualquer dos comandos a executar.

— **LIST DEF KEY** — é um complemento essencial do comando anterior que permite visualizar todas as teclas já definidas como teclas de função.

— **LIST FORMAT** — trata-se de um utilitário que permite configurar o display do programa actualmente a ser editado. Com este comando é possível configurar a listagem de um programa por forma a apresentar uma instrução por linha, o que torna mais fácil a leitura do programa.

CLOCK — permite aceder a um relógio em tempo real (com horas, minutos e segundos) que o **BETA BASIC** implementa. As funções que este relógio permite são bastante sofisticadas entre as quais se destaca um despertador que pode accionar o buzzer do **SPECTRUM**, e/ou desencadear um programa em **BASIC** ou código máquina, à hora de despertar. É também possível ler a hora do relógio e incluir essa informação nos programas.

— **RENUM** — permite renumerar, copiar e mover linhas de programa. É bastante útil quando utilizado com o comando **MERGE** para carregar subprogramas.

— **CLEAR** — permite redefinir o **RAMTOP** sem perder a informação das variáveis, ao contrário do que acontece com o **SINCLAIR-BASIC.**

Estes são alguns dos comandos que tornam o **BETA BASIC** tão atraente. No entanto as suas capacidades não se ficam por aqui, pois o seu ponto mais forte é o de permitir a programação estruturada.

DESENVOLVIMENTO DE PROGRAMAS

Com efeito, o **BETA BASIC 3.0** suporta a divisão de programas em subprogramas que realizam acções específicas e que tem um interface bem determinado com o programa principal. Esses subprogramas designam-se em programação por **procedures ou procedimentos.** As procedures aparecem já no **SUPER BASIC** do **QL**, e as do **BETA BASIC 3.0** são bastante parecidas com elas.

O interface das procedures umas com as outras e com o programa principal, é feito por meio de parâmetros das procedures, que são variáveis que tomam um determinado valor quando a respectiva procedure é invocada. A passagem de parâmetros no **BETA BASIC 3.0** pode fazer-se de duas maneiras distintas:

— **por valor** — em que o parâmetro toma o valor numérico que lhe é passado.

— por referência — em que o parâmetro representa o nome (ou referência) de uma dada variável que deve ser manipulada pela procedure. A passagem de parâmetro por referência é explicitamente indicada no **BETA BASIC 3.0** por **REF parâmetro**, à semelhança do **PASCAL** com a indicação **var parâmetro**. A passagem de parâmetros por referência é essencial quando se manipulam cadeias de caracteres e arrays.

Além de proporcionar um interface bem definido para as procedures, o **BETA BASIC 3.0** permite ainda definir variáveis locais das procedures, que só são conhecidas dentro das respectivas procedures tal como acontece nas outras linguagens de programação de alto nível.

Em complemento às procedures existem as funções que englobam todas as existentes no **SINCLAIR BASIC** e ainda outras específicas do **BETA BASIC 3.0**, tais como:

— conversão de grandezas numéricas de binário para decimal, de decimal para binário, de hexadecimal para decimal e de decimal para hexadecimal.

— identificação de subcadeias de caracteres dentro de outras cadeias de caracteres.

— conversão de cadeias de caracteres numéricas em inteiros.

— funções trignométricas mais rápidas que a implementação em **SINCLAIR BASIC.**

— formatação da impressão de números (com justificação à esquerda e à direita).

— conversão de caracteres de minúsculas para maiúsculas e vice-versa.

```
;
;TIMEX-SOFT
;
;ROTINA DE LEITURA DA DIRECTORIA DO TOS
;
;Retorna com o código de erro em A e BC (0 se tudo bem)
;
;
;
BUFDAT      EQU   2000H
BUFCOM      EQU   2100H
;
PUTDAT      EQU   0605H
PUTCOM      EQU   0608H
RESPOSTA    EQU   0626H
PAGEIN      EQU   0008H
PAGEOUT     EQU   0603H
;
RXSEC       EQU   27
;
;
RDDIR       PUSH  IY                  ;paginação
            LD    IY,0
            CALL  PAGEIN
            POP   IY
;
;leitura dos 16 sectores
;
            LD    DE,BUFFER           ;espaço para a directoria
            LD    B,0                 :contador
LOOP        CALL  RDSEC               ;leitura do sector
            LD    A,(BUFCOM+2)        ;código do erro
            AND   A
            JR    NZ,ERRO             ;erro na leitura
            PUSH  BC
            LD    HL,BUFDAT
            LD    BC,256              ;comprimento do sector
            LDIR                      ;transferir o sector
            POP   BC
            LD    A,B
            ADD   A,7                 ;skew de 7
            AND   00001111B           ;só interessam os bits 0 → 3
            LD    B,A
            JR    NZ,LOOP             ;até ler todos os sectores
;
            XOR   A                   ;não houve erros
            LD    B,0
            LD    C,A                 ;passagem do erro para o BASIC
ERRO        JP    PAGEOUT             ;paginação e retorno
;
RDSEC       PUSH  DE
            PUSH  BC
            LD    E,B                 ;número do sector a ler
            LD    D,4                 ;pista da directoria
            LD    C,0                 ;disco A
            LD    A,RXSEC             ;código do comando
            LD    (BUFCOM),A
            CALL  PUTCOM              ;executar o comando
            CALL  RESPOSTA            ;esperar resultados
            POP   BC
            POP   DE
            RET
;
BUFFER      DEFS  4096                ;comprimento da pista
;
            END
```

O TOS permite ler e escrever directamente para os sectores do disco. Estas funções são obviamente muito poderosas e devem ser usadas com as devidas precauções. A escrita no disco é irreversível e no caso de erro, a informação anterior fica permanentemente corrompida.

No exemplo escolhido o conteúdo da pista da directoria é lido para um buffer em memória. Esta rotina, por si só, não é de grande interesse, uma vez que se pode obter o mesmo resultado em BASIC, fazendo INPUT do canal 0. No entanto, abre a possibilidade de escrita na directoria, o que pode permitir, por exemplo, criar um utilitário que recupere um ficheiro acidentalmente apagado, repondo correctamente a entrada na directoria.

A estrutura da directoria é compatível com a do sistema operativo CP/M, que já se encontra instalado no FDD, e se espera seja em breve posto no mercado. Fica garantida, deste modo, a possibilidade de transferência de ficheiros entre os dois sistemas operativos.

3.d — Extensão do BASIC

O FDD-BASIC deixa aberta ao utilizador a possibilidade de acrescentar comandos ao BASIC do Spectrum.

Não existem limitações em relação à sintaxe dos comandos criados pelo utilizador, desde que sigam o aspecto normal dos comandos em BASIC. No entanto, recomenda-se que seja usada uma sintaxe que não entre em conflito com os outros comandos e preferencialmente que dê origem a um erro do interpretador logo com o primeiro código. Por exemplo os caracteres acessíveis por «SYMBL SHIFT» são uma boa escolha.

Para exemplificar esta possibilidade, apresentamos a seguir uma rotina que, depois de carregada em memória através de:

LOAD * "COMANDO.NEW" CODE ADDR

e executada com o comando:

RANDOMISE USER ADDR

onde ADDR é um endereço compatível com a memória disponível, por exemplo algures acima do RAMTOP, faz com que o Spectrum passe a aceitar, como sintacticamente correcto, o comando:

=

que fica sendo equivalente à sequência:

PRINT PAPER 0; INK 7; AT 10,8; "TESTE"

O comando apresentado não inclui parâmetros. Para os usar é necessário prolongar a análise sintáctica de modo a incluí-los e passar os valores para o módulo de execução, o que pode ser feito do mesmo modo que o Spectrum o faz e usando as rotinas deste.

```
;
;TIMEX-SOFT
;
;ROTINA DE EXTENSÃO DE BASIC
;
;
;Esta rotina é recolocável e tem 2 partes. A primeira prepara os vectores
;e a segunda executa o comando
;
;
;
CBAS        EQU   061DH
PAGEOUT     EQU   0603H
PAGEIN      EQU   0008H
;
VECTOR      EQU   213BH
CH_ADD      EQU   5C5DH
;
CH_OPEN     EQU   1601H
STMT_RET    EQU   1B76H
STMT_NE-
XT          EQU   1BF4H
;
NIVEIS      EQU   4
;
;alteração do vector de retorno do erro ao Spectrum
;
COMMAND     LD    HL,COMP          ;comprimento da primeira parte
            ADD   HL,BC            ;após USR XXXX BC tem o endereço
                                   ;de COMMAND e HL fica com o ende-
                                   ;reço absoluto da rotina.
;
            PUSH  IY               ;paginação
            LD    IY,0
            CALL  PAGEIN
            POP   IY
;
            LD    (VECTOR),HL      ;alterar a variável do sistema
            JP    PAGEOUT          ;retorno ao BASIC
;
COMP        EQU   $-COMMAND
;Ao chegar a RUNCOM o acumulador tem o código seguinte ao
;que originou o erro e está seleccionado o FDD-BASIC.
;
RUNCOM      LD    HL,(CH_ADD)      ;var. de sistema do Spectrum
            DEC   HL
            CP    0DH              ;fim da linha?
            JR    Z,SEMI_OK
            CP    ":"              ;mais comandos na linha?
            JR    Z,SEMI_OK
```

```
;
ERRO        INC   HL
            INC   HL                ;repõe o ponteiro
            LD    DE,11             ;local de retoma do processamento
                                    ;de erro
            PUSH  DE
            JP    PAGEOUT           ;paginação e devolução do erro
;
SEMI_OK     LD    A,(HL)
            CP    "="               ;será o nosso comando?
            JR    NZ,ERRO
;
;A partir deste ponto a sintaxe está correcta e falta repor o estado
;
TUDO_OK     LD    HL,0
            ADD   HL,SP             ;repor o stack para
            LD    BC,NIVEIS         ;a posição anterior ao Spectrum
            ADD   HL,BC             ;detectar o erro
            LD    SP,HL
;
;O Spectrum pode estar em modo de execução ou análise sintáctica.
;O teste é feito aqui.
;
            BIT   7,(IY+1)          ;teste do modo
            JR    NZ,RUN            ;executar o comando
;
;Estava em modo de análise sintáctica.
;
            LD    HL,STMT_NEXT      ;reentrada no interpretador
            PUSH  HL
            JP    PAGEOUT
;
;Vai executar o comando.
;
RUN         LD    A,2               ;canal de écran
            CALL  CBAS
            WORD  CH_OPEN           ;abre canal
            LD    HL,COMANDOS       ;ponteiro para a sequência
LOOP        LD    A,(HL)
            CP    "$"               ;fim da sequência
            JR    Z,FIM
            INC   HL
            PUSH  HL
            CALL  CBAS
            DEFW  PRINT_CH          ;rotina de escrita
            POP   HL
            JR    LOOP
;
FIM         LD    HL,STMT_END       ;fim do comando
            PUSH  HL
            JP    PAGEOUT
;
;
COMANDOS    DEFB  17,0              ;PAPER 0
            DEFB  16,7              ;INK 7
            DEFB  22,10,8           ;AT 10,8
            DEFM  "TESTE"
            DEFB  "$"               ;fim da sequência
;
            END
```

CONT. PRÓXIMO NÚMERO

Micro-análise

endereçamento do microprocessador. Ao ligar o computador 16KB de ROM são copiados para uma página de 16KB de RAM, sendo essa página seleccionada para figurar entre os 0 e os 16K no mapa de endereços. Ao seleccionar o modo 48K, três páginas são seleccionadas de modo a termos 48K de RAM, sendo o sistema de comutação de memória, desactivado.

No modo 128K são também selecionadas três páginas, ficando no entanto as rstantes quatro correspondentes a 64KB, acessíveis pelo sistema de comutação. Com este sistema consegue-se ter cerca de 104KB disponíveis para programas em Basic e 120KB para código máquina. De salientar que a memória extra é acedida em Basic pelos comandos LOAD!, SAVE!, CAT! e FORMAT!, usada de modo semelhante a um disco, daí a designação — RAM DISK

SOFTWARE

O 128K apresenta incorporado em ROM (nas primeiras versões em EPROM) o já conhecido ZX BASIC, o editor de texto, o calculador e o editor de ecrã de Basic.

Como software aplicável temos toda a biblioteca do ZX SPECTRUM 48K, dada a total compatibilidade e vários títulos que já estão a ser adaptados para o modo 128K, nomeadamente com assinaláveis melhoramentos em termos de som.

CRÍTICA

Como aspectos positivos convém realçar o editor de Basic, o excelente som, assim como o interface musical MIDI incorporado e a compatibilidade com o 48K.

Como aspectos negativos é de salientar o já referido dissipador a não existência de interface, paralelo Centronics o «jack» usado nos ports RS232 e MIDI serem do tipo do QL e difíceis de encontrar no mercado.

ESPECIFICAÇÕES TÉCNICAS

Microprocessador:
— Z80 A (3.5 MHz)

MEMÓRIA

— 128KB RAM em páginas de 16KBytes
— 32KB ROM, 16KBytes para modo 48K, 16KBytes para modo 128K

TECLADO

— Igual ao ZX Spectrum +
— KEYPAD numérico de 18 teclas

SOM

— processador AY-3-8912 de 3 canais
— som via TV

COMUNICAÇÃO

— Interface RS 232
— Interface musical MIDI
— MIC e EAR
— Saída RGB para monitor
— Saída vídeo composto
— conector de expansão

COMO TRABALHAR COM MATRIZES

O saber trabalhar com matrizes (arrays na linguagem anglo-saxónica), revela-se essencial para a elaboração de programas em Basic que apresentam uma certa complexidade. Isto porque a matriz se apresenta como a maneira mais simples de estruturar dados numéricos ou alfanuméricos, ficando a conhecer a sua localização, além de permitir gravar facilmente conjuntos de dados em cassette ou diskette para posterior consulta ou utilização.

As matrizes podem ter várias dimensões e número de elementos. Vamos introduzir o caso mais simples de uma matriz a uma dimensão e com 5 elementos numéricos.

A criação da matriz é feita pela instrução DIM, que permite ao computador reservar espaço na memória para guardar os dados pretendidos. DIM A (5) cria uma matriz numérica de uma dimensão e 5 elementos; de salientar que o nome da matriz não poderá ter mais que uma letra.

Ao ser introduzida agora a instrução LET A (1) = 3, e colocando o valor 3 na posição 1 da matriz, esquematicamente teremos:

A(1)	A(2)	A(3)	A(4)	A(5)
3				

Corra agora o seguinte programa introduzindo valores.

```
10 DIM A (5)
20 FOR n=1 TO 5
30 INPUT r
40 LET A (n)=r
50 NEXT n
```

Acabou assim de preencher todas as posições da matriz — se fizer agora PRINT A(1) verá aparecer no ecrã o valor introduzido na posição 1 e assim sucessivamente.

Um procedimento semelhante poder-se-á aplicar se desejar construir matrizes com mais de uma dimensão, sendo a sua criação executada também com a instrução DIM. Temos então como exemplo, a matriz A (3,5) de duas dimensões e A (3,5,5) de três dimensões e assim sucessivamente. Esquematicamente a representação da matriz A (3,5) será:

	1	2	3	4	5
1					
2					
3					

A introdução de valores efectua-se de modo semelhante ao utilizado para a matriz de uma dimensão, só que agora terão que ser introduzidas as duas coordenadas — temos como exemplo a instrução LET A (2,3) = 7, que introduz no quadrado correspondente a fila 2 e coluna 3 o valor 7; para as outras posições da matriz procede-se de modo semelhante.

Experimente agora fazer um pequeno programa que lhe permita preencher e ler sequencialmente todas as posições da matriz A (3,5) dimensionada.

De salientar que pode executar com os elementos da matriz todo o tipo de operações válidas para variáveis. Assim são válidas instruções como LET A (1) = A (1) + 1,

ou

PRINT A (1) + A (2) + A (3), etc.

O funcionamento de matrizes alfanuméricas baseia-se no mesmo princípio exposto para as numéricas, sendo um método extremamente eficaz para imprimir textos no ecrã.

O dimensionamento da matriz é feito usando a instrução DIM seguida de uma única letra e do caractere $; vamos exemplificar o seu funcionamento analisando o programa 1 em que é apresentada uma matriz a uma dimensão.

BASIC

sivamente na ecrã, formando neste caso, o conjunto das vogais.

As matrizes a duas dimensões podem ser encaradas como sendo conjuntos de palavras, o primeiro algarismo fixa o número de palavras e o segundo o número máximo de caracteres de cada uma delas.

Examine agora o programa 2 a que corresponde a matriz:

Na linha 10 a matriz é dimensionada com 24 elementos, na 20 são introduzidos os 24 caracteres correspondentes a cada elemento, nas linhas seguintes apresentam-se várias formas possíveis de imprimir elementos da matriz no ecrã. De salientar a instrução presente na linha 100 em que a soma de elementos corresponde a imprimi-los sucessivamente na ecrã, formando neste caso, o conjunto das vogais.

No programa 3 são aplicados vários conceitos expostos de modo a se conseguir uma pequena e simples base de dados.

PROGRAMA 1

```
  10 DIM a$(24)
  20 LET a$()="abcdefghijlmnopqr
stuvxz"
  30 PRINT a$()
  40 PRINT a$(1)
  50 PRINT a$(1 TO 5)
  60 PRINT a$(6 TO 10)
  70 PRINT a$(11 TO 15)
  80 PRINT a$(10 TO )
  90 PRINT a$( TO 15)
 100 PRINT a$(1)+a$(5)+a$(9)+a$(
14)+a$(20)
```

PROGRAMA 2

```
  10 DIM a$(4,11)
  20 LET a$(1)="REVISTA"
  30 LET a$(2)="SOFTFILE"
  40 LET a$(3)="ASUADE"
  50 LET a$(4)="INFORMATICA"
 100 PRINT AT 10,4;a$(2, TO 3)+a
$(3,1)+" "+a$(3,2 TO 4)+" "+a$(1
, TO 6)+a$(3,5 TO )
 110 PRINT AT 12,11;a$(4)
```

REVISTA
SOFTFILE
ASUADE
INFORMATICA

```
  10 DIM a$(100,20)
  20 DIM b$(100,10)
  22 PRINT AT 5,5;"1-LER DADOS"'
"      2-INTRODUZIR DADOS"
  23 PAUSE 0
  24 CLS : IF INKEY$="2" THEN GO
 TO 30
  25 LOAD "" DATA a$(): LOAD ""
DATA b$()
  30 DIM n$(20)
 100 PRINT AT 5,5;"NOME: ": INPU
T " STOP para acabar";n$()
 110 IF n$()=" STOP
       " THEN GO TO 500
 200 FOR n=1 TO 100
 205 LET a=n
 210 IF a$(n)=n$() THEN GO TO 30
0
 220 IF a$(n)="
   " THEN GO TO 400
 230 NEXT n
 300 PRINT AT 8,11;b$(a)
 310 PRINT AT 5,11;n$()
 320 INPUT "    Correcto S/N ?  "
;r$
 330 IF r$="n" THEN GO TO 410
 340 GO TO 30
 400 LET a$(a)=n$(): PRINT AT 5,
11;n$()
 410 PRINT AT 8,5;"TEL.: ": INPU
T t$
 420 LET b$(a)=t$: PRINT AT 8,11
;t$
 430 INPUT "    Correcto S/N ?  "
;r$
 440 IF r$="n" THEN GO TO 410
 450 PRINT AT 5,11;"
           "
 455 PRINT AT 8,11;"
           "
 460 GO TO 30
 500 CLS : PRINT AT 5,5;"1-PESQU
ISAR DADOS"''"        2-GRAVAR DADO
S"
 510 PAUSE 0
 520 CLS : IF INKEY$="1" THEN GO
 TO 30
 540 SAVE "dados1" DATA a$(): SA
VE "dados2" DATA b$()
 550 STOP
```

RESULTADOS:

Parabéns aos premiados e aos vencedores do concurso «KEOPS».

Parabéns aos que concorreram e não foram contemplados com lugares de honra.

Mais concursos iremos promover. Não será esta a única oportunidade.

No nosso próximo número iremos apresentar entrevista com todos os premiados.

1º ALEXANDRE G. RODRIGUES, premiado com 100.000$00

3º NÃO ATRIBUÍDO

2º EX-AEQUO

GRUPO: Pedro Coragem P. Fernandes
Paulo Jorge G. Lopes
Pedro Jorge A. A. da Veiga

GRUPO: Carlos Barroqueiro
Carlos Alberto
João Carlos
Maria Helena Barroqueiro

Cada um dos grupos premiados com 25.000$00

PREMIADOS DO MÊS

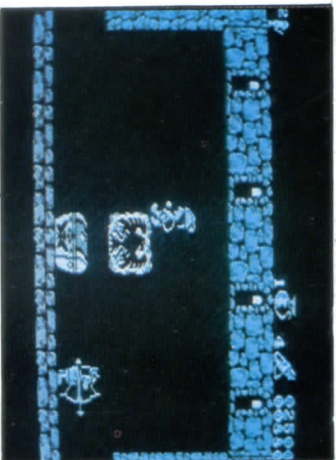
REVISTA Nº 2

JORGE AMARAL — Premiado com um computador TC 2048

REVISTA Nº 3

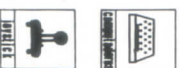
PAULO BERNARDO VARGAS — Premiado com um gravador TIMEX

SOFTWARE

Las Tres Luces de Glaurung

Lembram-se de um jogo intitulado «UNDERWURLDE» editado pela ULTIMATE? Pois este «LAS TRES LUCES DE GLAURUNG», da ERBE SOFT, é nem mais nem menos que uma versão espanhola do excelente programa da ULTIMATE.

Esta aventura decorre num labirinto constituído por vários andares, onde a mudança de piso se faz saltitando sobre diferentes objectos: quadros, prateleiras, brasões e outros. É dentro das arcas onde vamos encontrando que estão objectos importantes para levar o jogo a bom termo, nomeadamente chaves que facultam a passagem através de portas anteriormente fechadas. A dificultar a nossa movimentação estão os arqueiros inimigos, praticamente invencíveis, pois mal nos têm ao seu alcance, disparam as flechas: a melhor forma de os evitar é saltar sobre eles.

E mais não há a dizer sobre este jogo: os gráficos não são muito eficazes, e no que respeita aos «SPRITES» e aos movimentos do nosso Herói, aí este programa perde icomparavelmente para o «UNDER WURLDE».

Análise geral: Como qualquer cópia de valor muito inferior ao original, este programa é francamente desaconselhado.

SOFTWARE

Ghosts'n Goblins

A Elite volta a apostar na conversão de jogos de «COIN-MACHINES» para micro domésticos: A equipa que convertera o «COMMANDO» debruçou-se agora sobre «GHOSTS'N GOBLINS». A história é a de um cavaleiro medieval que tem de resgatar a sua bem-amada das mãos do rei das trevas, demónios e outra bicharada do género. O jogo principia no cemitério e atravessa diversas localizações (sempre recheadas de dificuldades) até atingir a aldeia fantasma onde fica a entrada para a cela da Princesa CATIVA.

O número inicial de vidas (9) é grande e ainda por cima, cada um vale por duas: ao primeiro choque com qualquer monstrinho, o susto apenas faz o cavaleiro perder a armadura (1); o segundo contacto é que é fatal, representando a perda de uma vida.

Nota-se que este jogo foi objecto de programação cuidada e tem a grande virtude de dosear correctamente o grau de dificuldade: os problemas postos em cada «SCREEN» exigem diferentes tácticas, pelo que o interesse do jogador se mantém intacto nas várias (muitas) tentativas de forminar este «GHOSTS'N GOBLINS».

POKES

NOTAS

POKES

NOTAS

UNI MICRO
microcomputadores
software
vídeos
Rua da Misericórdia, 67-1.º/Dto.
Telfs. 327073/362431
1200 Lisboa

SOFTWARE

Cobras Arc

Não há dúvida que a indústria espanhola de Software atingiu já um óptimo grau de desenvolvimento, bem patenteado aqui neste jogo e estratégia: em termos de programação este «COBRAS ARC» está perfeitamente ao nível do que de melhor se faz no Reino Unido.

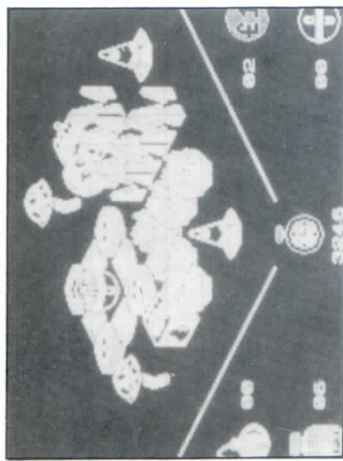
«COBRAS ARC» é um jogo de estratégia, com ilustrações estáticas (que aparecem na zona superior do écran), o texto é controlado por ícones. Nesta aventura, a nossa missão é encontrar e liquidar o príncipe Cobra, contando para tal com a força mística e a ajuda de objectos que recolhemos pelo caminho. É, ao fim e ao cabo, uma história igual a tantas outras...

No entanto, o que se realça neste programa da Dinamic SOFTWARE é o som, que sofreu um trabalho apuradíssimo, ao ponto de podermos ouvir o computador pronunciar todas as frases do texto, num espanhol de perfeita dicçaõ — uma verdadeira maravilha.

Análise geral: É um bom jogo de estratégia. A comprar.

SOFTWARE

microcomputadores software vídeos

Rua da Misericórdia, 67-1.º/Dto.
Telfs. 327073/362431
1200 Lisboa

Molecule Man

Em «Molecule Man», o objectivo é apanhar 16 circuitos que se encontram dispersos pelo labirinto do jogo: os circuitos são indispensáveis para que o MM possa activar o teletransportador e escapar do dédalo onde se encontra preso. Aqui estão as linhas com que se cose esta novíssima criação da MASTERTRONIC, que lança assim mais um jogo da moda, com labirinto tri-dimensional e gráficos cuidados ao ínfimo pormenor.

O jogo, por si só, não foge da mediocridade que tem caracterizado a maioria dos seus antecessores neste género. Mas a justificar os 7 meses que ROBIN THOMPSON gastou na elaboração do programa está a inclusaõ de um desenhador de labirinto, que permite ao jogador criar o seu próprio mapa de jogo. E não há dúvida que esta facilidade quase justifica a sua compra: os labirintos criados pelo utilizador podem ser guardados em cassette e posteriormente carregados, substituindo o mapa originalmente proposto pelo autor. Cada labirinto criado constitui um novo desafio — além disso, imaginem as trocas que podem fazer com os amigos!...

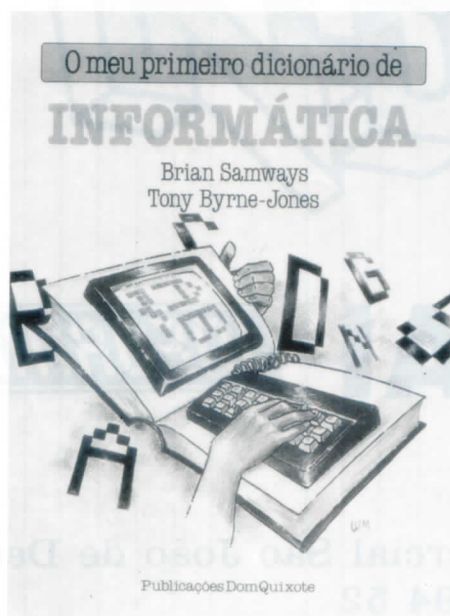
LITERATURA

TECNOLOGIAS DA INFORMAÇÃO

Ao longo deste livro Helder Coelho aborda os vários aspectos que caracterizam as modernas tecnologias da informação ou dos três C: Computador, Controlo e Comunicação.

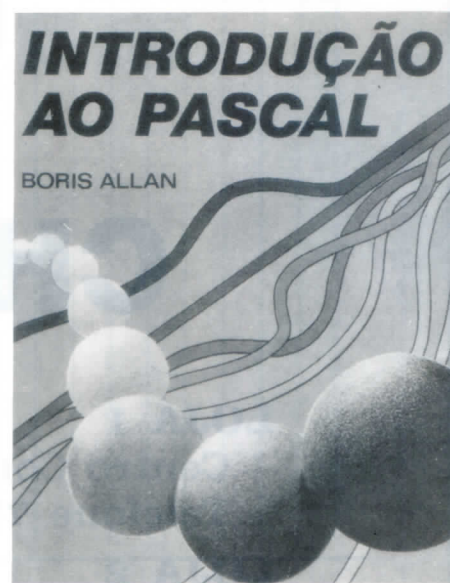
É analisado de uma forma sistemática o significado da introdução destas tecnologias a nível nacional e internacional e o seu impacto sobre os processos produtivos, a indústria, a investigação e a própria sociedade.

Um livro absolutamente indispensável a quem quer estar a par das mutações sociais, económicas e tecnológicas introduzidas pelas novas tecnologias da informação, das orientações associadas ao seu desenvolvimento e das perspectivas para a sua aplicação em Portugal.

Título: Tecnologias da Informação
Autor: Helder Coelho
Colecção: Universidade Moderna
Editora: Publicações D. Quixote
Nº de páginas: 247

O MEU PRIMEIRO DICIONÁRIO DE INFORMÁTICA

O mundo dos computadores possui uma linguagem constituída por termos técnicos e palavras em gíria específica, que com a vulgarização dos microcomputadores de uso doméstico urge conhecer. **"O MEU PRIMEIRO DICIONÁRIO DE INFORMÁTICA"** escrito por dois autores profundamente ligados às áreas da informática e do ensino, constitui um guia indispensável, de grande clareza e conteúdo didáctico, para aprendizagem das palavras necessárias ao uso do microcomputador.

Título: O meu primeiro dicionário de informática.
Autores: Brian Samways & Tony Birne-Jones
Editora: Publicações D. Quixote
Nº de páginas: 102

INTRODUÇÃO AO PASCAL

O Pascal constitui sem dúvida, a seguir ao BASIC, a linguagem de programação mais popular.

Este livro é um bom elemento de aprendizagem tanto para o leitor que contacta pela primeira vez com esta linguagem como para o leitor mais experiente, dando-lhes uma compreensão mais profunda do Pascal com especial ênfase para os seus aspectos estruturais.

Título: Introdução ao Pascal
Autor: Boris Allan
Colecção: Biblioteca de informática
Editora: Verbo
Nº de páginas: 191

LITERATURA

102 PROGRAMAS DE JOGOS PARA O AMSTRAD

Os jogos continuam a constituir, principalmente junto das camadas mais jovens, o principal objectivo na aquisição de um microcomputador. Sendo assim a introdução de jogos poderá servir como incentivo para um jovem aprender a programar. Visto nesta perspectiva, este livro, com todos os seus programas descritos e analisados linha a linha, será um bom guia para a progressiva exploração do BASIC do Amstrad.

Título: 102 programas de jogos para o Amstrad
Autor: Jacques Deconchat
Colecção: Arte de Viver
Editora: Publicações Europa--América
Nº de páginas: 142

Joy II

A utilização do joystick permite, principalmente ao consumidor de jogos, um controlo mais preciso da acção imprimindo simultaneamente uma maior emoção à maioria dos jogos. Evita também uma excessiva solicitação do teclado, normalmente exigida pelos jogos de acção e causa frequente de inúmeras avarias.

Apesar dos vários microcomputadores já incorporarem um porto de joystick, alguns ainda não o possuem, entre eles o ZX Spectrum, certamente o microcomputador mais utilizado no nosso país. Uma maneira de contornar esta situação reside na compra de um interface apropriado que possibilite a ligação dos vários joysticks existentes no mercado. A JG Componentes do Porto possui no mercado o interface adequado a esta função, trata-se do JOY II.

O JOY II é um interface que permite, depois de ligado ao conector traseiro do ZX Spectrum/TC 2048/TC 2068, o controlo por joystick de todos os jogos que possuam a opção Kempston ou Quickshot.

Utilização

A utilização do interface é extremamente fácil, é no entanto importante que o leitor cumpra uma série de regras básicas, algumas delas válidas para o ligar de qualquer interface, e sem as quais poderá causar danos irreversíveis no seu microcomputador.

Antes de ligar o JOY II convém verificar se este possui na ficha um barramento plástico, se não, não o utilize e reclame junto do revendedor, tal pormenor é importante pois o barramento que vai obrigar o interface a ser inserido de uma forma correcta. Ao ligar o interface ao conector traseiro assegure-se que o microcomputador está desligado, a não observância desta regra é a causa mais frequente de avarias na CPU. Depois do interface correctamente ligado, ligue o joystick a uma das suas saídas, de salientar que estas funcionam em paralelo, pode então ligar o computador e carregar o jogo da forma habitual, escolha a opção Kempston ou Quickshot e divirta-se.

Utilização nos seus próprios programas

Os utilizadores que desejem usar o joystick nos seus próprios programas, podem fazê-lo de uma forma extremamente simples bastando para tal incluir no seu programa a seguinte rotina:

```
BASIC: LET A=IN 31
       IF A=1 direita
       IF A=2 esquerda
       IF A=4 baixo
       IF A=8 cima
       IF A=16 fogo

ASSEMBLER: IN A, 31
           IF A=1H
           IF A=2H
           IF A=4H
           IF A=8H
           IF A=10H
```

O interface pode ainda ler as quatro posições intermédias correspondentes a : Noroeste, Nordeste, Sudoeste e Sudeste. O valor de A obtém-se nestes casos da soma dos valores das posições adjacentes.

De salientar a presença de um botão de reset que possibilita não ser necessário desligar o computador ao querer mudar o jogo ou no caso de «crash» do programa.

Apresentação

O interface apresenta-se em caixa de ABS preto. É provido de um conector na sua parte traseira, e um botão de reset na face superior onde se encontram também as duas fichas de ligação do joystick. É também fornecida uma folha de instruções possuindo todas as indicações necessárias a uma correcta utilização.

Comentários:

Não experimentámos qualquer dificuldade na utilização do JOY II, que funcionou sempre correctamente, no entanto o barramento plástico apresenta-se bastante frágil pelo que é necessário o maior cuidado para que não se parta ou solte após algumas utilizações.

Monitor Slot

O recente aparecimento em Portugal de monitores a preços extremamente acessíveis permite abandonar de vez a utilização de televisores com microcomputadores, utilização esta particularmente cansativa ao serem usados programas em 64 colunas.

O uso de monitores com alguns microcomputadores, nomeadamente o ZX Spectrum revela-se problemática, dada a não existência de saída de vídeo composto (tal assunto já foi abordado na Softfile nº 1) uma solução consiste em aquirir um interface que permita esta ligação.

O interface actualmente presente no mercado para este fim é o MONITOR SLOT da JG Componentes, que pode ser ligado ao conector traseiro de qualquer computador da linha Timex/Sinclair e apresenta uma saída de vídeo composto para monitor e um botão de reset. O MONITOR SLOT dispõe também de um conector na sua parte traseira de modo a possibilitar a ligação de outros periféricos.

Apresentação

O MONITOR SLOT apresenta-se como um pequeno interface em ABS preto, possuindo na sua face superior a saída para monitor, o botão de reset e uma placa metálica que lhe confere maior solidez.

INTERFACES PARA ZX SPECTRUM/+, TIMEX TC 2048/2068

TODOS OS NOSSOS INTERFACES TEM GARANTIA DE 6 MESES E ASSISTÊNCIA TÉCNICA. BOTÃO DE RESET E POSSIBILIDADE DE EXPANSÃO, SÃO CARACTERÍSTICAS COMUNS A TODOS OS NOSSOS INTERFACES.

INTERFACE JOYSTICK PROGRAMÁVEL

Permite programar qualquer que sejam as teclas de comando do jogo, compatível com ZX Spectrum/+.

INTERFACE SOM->TV

Basta só encaixar na parte de trás do seu ZX Spectrum/+, TIMEX TC 2068 e terá os sons do computador reproduzidos no televisor.

INTERFACE LIGHTPEN

Permite-lhe com uma caneta especial executar desenhos no ecran do seu televisor, compatível com ZX Spectrum/+, TIMEX TC 2048/2068.

MONITORSLOT

Permite-lhe ligar directamente no seu ZX Spectrum/+ um monitor qualquer existente no mercado, reproduzindo imagem monocromática com qualidade, assim como executar o RESET e inclui uma pequena luz de cor vermelha para indicar LIGADO.

JOY II

É um interface para joystick do tipo KEMPSTON / QUIKSHOOT com duas saídas para joystick assim como botão de RESET e expansão para outros periféricos, inclui a oferta de uma case plástica, e já disponível no mercado.

FIT 1

Ficha em forma de T para ligar dois periféricos em paralelo, no computador ZX Spectrum e TIMEX TC 2048

FIT 2

Ficha em forma de T para ligar dois periféricos em paralelo no computador TIMEX TC 2068

J.G. COMPONENTES

R. PASSOS MANUEL 223 LOJA 24 • 4000 PORTO • TEL. 38 32 25
Nestes preços inclui IVA e despesas de transporte.

PARA J.G. COMPONENTES
R. PASSOS MANUEL 223 L. 24 • 4000 PORTO

QUANTIDADE	DESCRIÇÃO	PREÇO	TOTAL
	Joystick Programável	4400$00	
	Som—>TV	3900$00	
	Lightpen	3700$00	
	Monitor Slot	2200$00	
	Joy II	2200$00	
	FIT 1	1750$00	
	FIT 2	1750$00	
		TOTAL:	

Envio o cheque / Vale de Correio no valor de: ____________

NOME ____________

MORADA ____________

TELEFONE ____________

J.G. COMPONENTES • PORTO
TEL.: 38 32 25

CONDIÇÕES PARA COMERCIANTES. CONTACTE-NOS.

PROGRAMAS

O programa "Fisco" foi elaborado com base no Cód. do Imposto Complementar e nos Decretos que introduziram alterações e actualizam algumas das suas disposições.

Este programa é destinado ao cálculo normal do imposto, não tendo em consideração o caso especial respeitante aos rendimentos do trabalho da Função Pública.

Atendendo à evidente complexidade do sistema fiscal, de que este imposto é exemplo, foram considerados neste programa os casos da grande maioria dos contribuintes, pelo que, julgamos ser de grande actualidade e utilidade a sua utilização.

INSTRUÇÕES:

a) Rendimentos do trabalho, reformas, pensões, etc
b) Red. de prédios rústicos ou urbanos. (Habitação própria, etc.).
c) A utilizar quando o conjuge trabalha ou tem rendimentos como em b).
d) Introduzir o número de filhos ou 0.
e/f) Deduções nos termos dos Art. 28/30, (juros, seguros de vida, desp. de saúde, quotizações — sind./caixa).

Nota: Introduza os valores em escudos (ex. 10 contos = 10000) ou na sua inexistência apenas um 0.

```
   8 REM * Versao 2     *
  10 REM   * ECRAN *
  15 BORDER 2: PAPER 2: INK 6: C
LS
  20 DRAW 255,0: DRAW 0,175: DRA
W -255,0: DRAW 0,-175
  25 PRINT AT 5,7;"PROGRAMA ""FI
SCO"""
  30 PRINT INVERSE 1;AT 12,5;" ©
 Mario Mateus - 1986 "
  35 PLOT 0,23: DRAW 255,0
  40 PRINT AT 20,1;" ";
  42 RESTORE
  45 FOR X=1 TO 26
  50 READ logo
  55 PRINT CHR$ logo;: BEEP 0.05
,logo/2
  60 NEXT X
  65 DATA 32,67,65,76,67,85,76,7
9,32,73,77,80,46,32,67,79,77,80,
76,69,77,69,78,84,65,82
  75 PRINT #0;AT 1,1;"Qualquer t
ecla para continuar"
  80 PAUSE 0
  85 BORDER 5: PAPER 5: INK 0: C
LS
  95 PRINT FLASH 1;AT 2,10;"INST
RUCOES"
 100 PRINT AT 4,1;"O programa FI
SCO calcula o Im-";" posto Compl
ementar. Introduza ";" os dados 
pela ordem solicitada";" pelo co
mputador.Posteriormente";" o cal
culo e realizado automati";" -ca
mente. Depois pague com um ";" s
orriso nos labios."
 105 PRINT AT 18,16;"BOA SORTE"
 120 PRINT #0;AT 1,1;"Qualquer t
ecla para continuar"
 126 PAUSE 0
 127 CLS
 130 REM * DADOS *
 132 LET u$="#00"
 135 PRINT INVERSE 1;AT 0,2;"**
RENDIMENTOS E DEDUCOES **"
 140 PRINT AT 1,0;"-1.TITULAR:"
 145 INPUT #0;"RENDIMENTOS:  ";a
 150 PRINT AT 2,2;"RENDIMENTOS
     ";AT 2,(28-LEN STR$ a);a;u$
 155 INPUT #0;"IMP.PROF.:  ";b
 160 PRINT AT 3,2;"IMP.PROFISSIO
NAL   ";AT 3,(28-LEN STR$ b);b;u$
 165 INPUT #0;"F.DESEMP.:  ";c
 170 PRINT AT 4,2;"F.DESEMPREGO
     ";AT 4,(28-LEN STR$ c);c;u$
 175 INPUT #0;"OUTROS RENDIM.  "
;d
 180 PRINT AT 5,2;"OUTROS RENDIM
     ";AT 5,(28-LEN STR$ d);d;u$
 185 INPUT #0;"CASADO (S/N)?  ";
e$
 190 IF e$="N" OR e$="n" THEN LE
T x=180000: GO TO 400
 195 IF e$="S" OR e$="s" THEN LE
T x=360000: GO TO 200
 196 IF e$<>"N" AND e$<>"n" OR e
$<>"S" AND e$<>"s" THEN GO TO 18
5
 200 PRINT AT 7,0;"-2.TITULAR:"
 205 INPUT #0;"RENDIMENTOS:  ";f
 210 PRINT AT 8,2;"RENDIMENTOS
     ";AT 8,(28-LEN STR$ f);f;u$
 215 INPUT #0;"IMP.PROF.:  ";g
 220 PRINT AT 9,2;"IMP.PROFISSIO
NAL   ";AT 9,(28-LEN STR$ g);g;u$
 225 INPUT #0;"F.DESEMP.:  ";h
 230 PRINT AT 10,2;"F.DESEMPREGO
     ";AT 10,(28-LEN STR$ h);h;
u$
 235 INPUT #0;"OUTROS RENDIM.:
";i
 240 PRINT AT 11,2;"OUTROS RENDI
M.     ";AT 11,(28-LEN STR$ i);i;
u$
 255 PRINT AT 12,0;"-DEDUCOES:"
 2[illegible] INPUT #0;"DEPENDENTES (S/N)
? ";j$
 260 IF j$="N" OR j$="n" THEN GO
TO 310
 26[illegible] IF j$[illegible] OR j$="s" THEN GO
TO 266
 265 IF j$<>"N" OR j$<>"n" AND j
$<>"S" OR j$<>"s" THEN GO TO 255
 266 PRINT AT 13,1;"DEPENDENTES:
"
 270 INPUT #0;"No. FILHOS (-11 a
nos): ";l
 275 IF l=0 THEN GO TO 290
 280 IF l>=1 THEN LET l=l*40000
 285 PRINT AT 14,2;"-FILHOS (-11
anos)";AT 14,(28-LEN STR$ l);l;
u$
 290 INPUT #0;"No. FILHOS (+11 a
nos): ";m
 295 IF m=0 THEN GO TO 310
 300 IF m>=1 THEN LET m=m*60000
 305 PRINT AT 15,2;"-FILHOS (+11
anos)";AT 15,(28-LEN STR$ m);m;
u$
 310 INPUT #0;"SINDICATO/CAIXA:
";n
 315 PRINT AT 16,2;"-SINDICATO/C
AIXA";AT 16,(28-LEN STR$ n);n;u$
 320 INPUT #0;"JUROS/OUT.DEDUCOE
S: ";o
 325 PRINT AT 17,2;"-JUROS/OUT.D
EDUC.";AT 17,(28-LEN STR$ o);o;u
$
 326 LET Q=a+d+i+f
 327 LET s=f*30/100
 328 IF s>105000 THEN LET s=1050
00
 331 LET p=a*30/100
 332 IF p>105000 THEN LET p=1050
00
 333 IF j$="S" OR j$="s" THEN GO
TO 335
 334 IF j$="N" OR j$="n" THEN GO
TO 336
 335 LET R=b+c+g+h+l+m+n+o+x+p+s
: GO TO 340
 336 LET R=b+c+g+h+n+o+x+p+s
 340 PRINT AT 19,8;"* T O T A I
S *"
 345 PRINT AT 20,2;"RENDIMENTOS.
.:.";AT 20,(28-LEN STR$ Q);Q;u$
 350 PRINT AT 21,2;"DEDUCOES....
..:";AT 21,(28-LEN STR$ R);R;u$
 355 GO TO 640
 400 PRINT AT 12,0;"-DEDUCOES:"
 410 INPUT #0;"DEPENDENTES (S/N)
? ";j$
 420 IF j$="N" OR j$="n" THEN GO
TO 530
 430 IF j$="S" OR j$="s" THEN GO
TO 440
```

CLUBE LEITORES SOFTFILE

MERCADO DE SOFTWARE

A revista SOFTFILE em colaboração com as firmas CHAI-CHAI e UNIMICRO oferece, aos seus leitores, a partir deste número um serviço de vendas de Software (programas/jogos).

A preços extremamente baixos e com a garantia de qualidade enviamos para qualquer parte do País (máx. 1 semana), programas em cassette ou diskette.

Programas em cassette:
1 programa: 200$00 + 100$00 para despesas envio pelo correio
6 programas: 1000$00 + 100$00 para despesas envio pelo correio

Programas em diskette:
Cada diskette com 4 programas à sua escolha 2500$00 + 100$00 para despesas envio pelo correio.
3 Diskettes 6500$00 + 100$00 para despesas envio pelo correio.

3 WEEKS IN PARADISE
911 TS
1942
2112 AD
ACADEMY
ACTION REFLEX
ALCHEMIST
ALI BABA
ALLIEN 8
ALIENS
AIRWOLF
ANT ATTACK
ANTIRIAD
ARC OF YESOD
ASTERIX
ASTRO CLONE
ATTACK OF KILLER TOMATOES
AVALON
AVENGER
BABALIBA
BALLBLAZER
BALLOONING
BATMAN
BIGGUES
CAMELOT WARRIORS
CAULDRON I
CAULDRON II
CHEQUERED FLAG
CHIMENA
CHUCKIE EGG 1
COBRA
COBRA'S ARC
COLOSSUS 4 CHESS
COMBAT LYNX
CORE
CYCLONE
CYCLO
CYRUS CHESS I
CYRUS CHESS II
DAMBUSTERS
DAN DARE
DANDY
DRAUGHTS
DRUID
DUSTIN
DYNAMITE II
ENIGMA FORCE
EQUINOX
FAIRLIGHT
FANTASTIC VOYAGE
FAIRLIGHT 2
FIGHTER PILOT
FRED
FROST BYTE
FUTURE KNIGHT
GALVAN
GAUTLET
GLADIATOR
GLIDER RIDER
GIFT FROM THE GODS
GOONIES
DANDY
KNIGHTMARE RALLY
THE GREAT ESCAPE
GRAND NATIONAL
GREEN BENET
GYROSCOPE
GUN FRIGHT
HEARTLAND
HEAVY OR MAGIC
ICICLE WORKS
INCUPS
INCREDIBLE FIREMAN
PRODIGY
STRIKE FORCE COBRA
MAG MAG
SACRED ARMOUR OF ANTIRIAD
INFILTRATOR
HIGHLANDER
BREAKTHRU
IT'S A KNOCKOUT
SHOCKWAVE RIDER
FIRELORD
1942
JAILBREAK
DEACTIVATOR
CYBORG
SNOOKER
DODGY GEEZERS
MASTER OF UNIVERSE
BOMBJACK II
AIRWOLF
SPACE HARRIER
SCOOBY DOO
DOUBLE TAKE
BAZOOKA BILL
SHAO LIN'S ROAD
WORLD GAME
INFILTRATOR
JET SET WILLY
KIREL
KOKOTONI WILF
KNIGHT LORE
KNIGHT RIDER
KNIGHT TIME
LAS TRES LUCES DE GLAURUNG
LAZY JONES
LEGEND OF AMAZON WOMEN
LIGHT FORCE
LOCOMOTION
MACHINE CODE FOR ENGINEERS
MACHINE CODE TUTOR
MAGIC CARPET
MANICMINER
MARBLE MADNESS
MARSPORT
MANTRONIX
MATCH DAY
MATCH POINT
MERMAID MADNESS
MOLECULE MAN
MOVIE
MS PALMAN
NEXOR
NIGHT SHADE
NONAMED
NOSFERATU
OLE TORD
OLIE AND LISSA
PAPER BOY
PENETRATON
PENTAGRAM
PHANTOMAS I
PHANTOMAS II

PING PONG
PRODIGY
PROFANATION
PSION CHESS 3D
PYRACURSE
QUAZATRON
RAMBO
RASPUTIN
REVOLUTION
ROBIN OF THE WOOD
ROCKY HORROR SHOW
SABOTEUR
SABRE WULF
SAI COMBAT
SCOOBY DOO
SCUBA DIVE
SGUISAM
SHOW JUMPING
SILENT SERVICE
SIR FRED
SKY FOX
SKOOL DAZE
SORCERY
SPACE HARRIER
SPELLBOUND
SPIKY HAROLD
SPITFIRE 40
SPY HUNTER
STAINLESS STEEL
STAR GUDER
STRAR STRIKE II
STREET HAWK
STRIKE FORCE HARRIER
SUBTERRANIEN STRYKER
SUPERCHESS 3.5
SWEEVO'S WORD
TANTALUS

ALGUNS EXEMPLOS DE PROGRAMAS EM CASSETTE

PAPER BOY
3 WEEKS IN PARADISE
MISSION OMEGA
W.A.R.
HEAVY ON THE MAGICK
LEADER BOARD
PSI-5 TRADING CO.
LAW OF THE WEST
DRAGON'S LAIR
DAN DARE
LIGHTFORCE
POWER PLAY
DARK SCEPTRE
T.T. RACER
UCHI MATA
REVOLUTION
HARRIER
FIST II
HEARTLAND
DYNAMITE DAN II
MIAMI VICE
TRIVIAL PURSUITS
THEATRE EUROPE
ACE
G. GOOCH'S CRICKET
FOOTBALL MANAGER
GHOST & GOBLIN
GREEN BERET
ELITE
THE FALL GUY
THE GREAT ESCAPE
THINK
TIME GATE
TIME TRAX
TIRNANOG
TOMMY
TOP GUN
TRAP DOOR
UNDER WULDE
UNIVERSAL HERO
URIDIUM
VIDEO POOL
WHAM! MUSIC BOX
GRAPHIC ADVENTURE CREATOR
KNIGHT RIDER
DRUID
GALVAN
SURF CHAMP
BOUNCES
WORLD CUP FOOTBALL
SPIDERMAN
POLE POSITION
SKYFOX
ASTERIX
QUAZATRON
MONOPOLY
URIDIUM
HARDBALL
TARZAN
TAUCETI
TENNIS
TERRA CRESTA
THANATOS
FOOTBALLER OF THE YEAR
FUTURE KNIGHT
WAY OF THE TIGER II (THE ADVENGER)
TRAIL BLAZER

Envie o seu pedido juntamente com a importância em vale postal ou cheque para a Revista Softfile — R. Rodrigo da Fonseca, 95-4.º, 1200 Lisboa

NOME

MORADA

TÍTULOS CASSETTE	VALOR	TÍTULOS DISKETTE	VALOR
1		1	
2		2	
3		3	
4		4	
5		5	
6		6	
7		7	
8		8	
TOTAL		TOTAL	

ANTÓNIO SARGENTO

LETRAS GREGAS

ZX SPECTRUM • TC 2048 • TC 2068

750 ESC

```
 435 IF j$<>"N" OR j$<>"n" AND j
$<>"S" OR j$<>"s" THEN GO TO 410
 440 PRINT AT 13,1;"DEPENDENTES:
"
 450 INPUT #0;"No. FILHOS (-11 a
nos): ";l
 460 IF l=0 THEN GO TO 490
 470 IF l>=1 THEN LET l=l*40000
 480 PRINT AT 14,2;"-FILHOS (-11
anos)";AT 14,(28-LEN STR$ l);l;
u$
 490 INPUT #0;"No. FILHOS (+11 a
nos): ";m
 500 IF m=0 THEN GO TO 530
 510 IF m>=1 THEN LET m=m*60000
 520 PRINT AT 15,2;"-FILHOS (+11
anos)";AT 15,(28-LEN STR$ m);m;
u$
 530 INPUT #0;"SINDICATO/CAIXA:
";n
 540 PRINT AT 16,2;"-SINDICATO/C
AIXA";AT 16,(28-LEN STR$ n);n;u$
 550 INPUT #0;"JUROS/OUT.DEDUCOE
S: ";o
 560 PRINT AT 17,2;"-JUROS/OUT.D
EDUC ";AT 17,(28-LEN STR$ o);o;u
$
 570 LET Q=a+d
 580 LET p=a*30/100
 590 IF p>105000 THEN LET p=1050
00
 593 IF j$="S" OR j$="s" THEN GO
 TO 600
 595 IF j$="N" OR j$="n" THEN GO
 TO 605
 600 LET R=b+c+l+m+n+o+x+p: GO T
O 610

 605 LET R=b+c+n+o+x+p
 610 PRINT AT 19,8;"* T O T A I
S *"
 620 PRINT AT 20,2;"RENDIMENTOS.
....";AT 20,(28-LEN STR$ Q);Q;u$
 630 PRINT AT 21,2;"DEDUCOES....
....";AT 21,(28-LEN STR$ R);R;u$
 640 INPUT #0;"DESEJA IMPRESSORA
(S/N)?";t$
 650 IF t$<>"S" AND t$<>"N" THEN
 GO TO 640
 670 IF t$="S" THEN COPY
 750 CLS : PRINT INVERSE 1;AT 0,
1;"CALCULO DO IMPOSTO COMPLEMENT
AR"
 760 PRINT AT 3,12;"** 1985 **"
 770 PRINT AT 5,2;"-RENDIMENTOS.
....";AT 5,(28-LEN STR$ Q);Q;u$
 780 PRINT AT 7,2;"-DEDUCOES....
....";AT 7,(28-LEN STR$ R);R;u$
 790 PRINT AT 9,20;"__________
"
 800 LET K=Q-R
 801 IF Q<R THEN GO TO 5005
 810 PRINT AT 11,2;"REND.COLECTA
VEL";AT 11,(28-LEN STR$ K);K;u$
 820 LET z=K
 830 IF e$="N" OR e$="n" THEN GO
 TO 2020
 850 REM * casados *
 900 IF z<=320000 THEN LET z=z*4
/100: GO TO 5000
 910 IF z>320000 AND z<=630000 T
HEN LET z=(z*6/100)-6400: GO TO
5000
 920 IF z>630000 AND z<=1040000
THEN LET z=z*8/100-19000: GO TO
5000
 930 IF z>1040000 AND z<=1550000
 THEN LET z=(z*12/100)-60600: GO
 TO 5000
 940 IF z>1550000 AND z<=1900000
 THEN LET z=(z*18/100)-153600: G
O TO 5000
 950 IF z>1900000 AND z<=2700000
 THEN LET z=(z*24/100)-267600: G
O TO 5000
```

```
 960 IF z>2700000 AND z<=3530000
 THEN LET z=(z*30/100)-429600: G
O TO 5000
 970 IF z>3530000 AND z<=4450000
 THEN LET z=(z*36/100)-641400: G
O TO 5000
 980 IF z>4450000 AND z<=5250000
 THEN LET z=(z*42/100)-908400: G
O TO 5000
 990 IF z>5250000 AND z<=6050000
 THEN LET z=(z*48/100)-1223400:
GO TO 5000
1000 IF z>6050000 THEN LET z=(z*
50/100)-1344400: GO TO 5000
2020 REM * solteiros *
2030 IF z<=270000 THEN LET z=(z*
4.8/100): GO TO 5000
2040 IF z>270000 AND z<=520000 T
HEN LET z=(z*7.2/100)-6480: GO T
O 5000
2050 IF z>520000 AND z<=870000 T
HEN LET z=(z*9.6/100)-18960: GO
TO 5000
2060 IF z>870000 AND z<=1210000
THEN LET z=(z*14.4/100)-60720: G
O TO 5000
2070 IF z>1210000 AND z<=1560000
 THEN LET z=(z*21.6/100)-147840:
 GO TO 5000
2080 IF z>1560000 AND z<=2240000
 THEN LET z=(z*28.8/100)-260160:
 GO TO 5000
2090 IF z>2240000 AND z<=2960000
 THEN LET z=(z*36/100)-421430: G
O TO 5000
3000 IF z>2960000 AND z<=3650000
 THEN LET z=(z*43.2/100)-634560:
 GO TO 5000
3010 IF z>3650000 AND z<=4400000
 THEN LET z=(z*50.4/100)-897350:
 GO TO 5000
3020 IF z>4400000 AND z<=5000000
 THEN LET z=(z*57.6/100)-1214140
: GO TO 5000
3030 IF z>5000000 THEN LET z=(z*
60/100)-1334150
5000 PRINT INVERSE 1;AT 15,2;"IM
POSTO A PAGAR:"; INVERSE 0;AT 15
,(28-LEN STR$ z);INT z;u$
5003 GO TO 5010
5005 PRINT INVERSE 1;AT 15,2;"IM
POSTO A PAGAR:"; FLASH 1;AT 15,2
0;"ISENTO"
5010 INPUT #0;"DESEJA IMPRESSORA
 (S/N)?";t$
5020 IF t$<>"S" AND t$<>"N" THEN
 GO TO 5010
5030 IF t$="S" THEN COPY
5050 INPUT #0;"DESEJA OUTRO CALC
ULO (S/N)?";a$
5060 IF a$<>"S" AND a$<>"N" THEN
 GO TO 5050
5070 IF a$="S" THEN GO TO 10
5080 IF a$="N" THEN STOP
9999 SAVE "fisco" LINE 1
```

O Spectrum 48k tem os seus caracteres definidos num lugar específico da memória. Eles estão definidos na ROM a partir do byte 15616 até ao 16383. Cada caractere está nesta parte da memória de oito em oito bytes.

Esta parte da memória é controlada pela CHARS (sistema de variáveis), que tem o endereço do primeiro caractere menos 256(15360). Se definirmos uma nova memória de caracteres na RAM, a apontarmos para lá da CHARS, então teremos o Spectrum a fazer novos caracteres.

Para menos trabalho pode-se gravar os bytes, depois de correr o programa (SAVE "NOME" CODE XXXXX,768), e apenas mudar a CHARS.

Ou ainda com um pequeno programa em linguagem máquina para mudar a CHARS.

```
LD HL, XXXXX 21nins
LD (23606),HL 22365C
RET          C9
```

Total de bytes: 7

Junta-se este programa aos bytes (8 bytes menos que o endereço da memória de caracteres definida por ti) e grava-se:

SAVE "NOME" CODE XXXXX - 8,776.

O programa em Basic para introduzir a linguagem máquina poderá ser (pode utilizar estas linhas no programa principal):

```
1 FOR n=XXXXX-8 TO
XXXXX-1
2 RESTORE 6
3 READ a
4 POKE n,a
5 NEXT n
6 DATA 33,ni,ns,34,54,92,201
```

Depois de tudo gravado para

PROGRAMAS

introduzir bastará LOAD"" CODE e RANDOMIZE USR XXXXX-8.

NOTA: O endereço da nova memória de caracteres que se deve "pokar" na CHARS, é sempre o primeiro byte menos 256.

Não esquecer que a CHARS é constituída por dois bytes, o primeiro para a componente inferior (ni), e o segundo para a componente superior (ns) do endereço (XXXXX).

Entre parêntesis estão as variáveis que se utilizou acima e que irão ser definidas por si).

```
  25 RESTORE 4000
  30 FOR a=15616 TO 16376 STEP 8
  40 IF a=15616 AND a=15872 THEN
GO TO 140
  50 IF a=16088 AND a=16128 THEN
GO TO 140
  60 IF a=16344 AND a=16376 THEN
GO TO 140
  70 FOR m=0 TO 7
  80 READ f
  90 POKE a+m+20000,f
 100 NEXT m: MEXT a
 130 GO TO 9000
 140 FOR m=0 TO 7
 150 POKE 20000+m+a,PEEK (a+m)
 160 NEXT m: NEXT a
4000 DATA 0,60,66,66,126,66,66,0
4001 DATA 0,124,66,124,66,66,124
,0
4002 DATA 0,66,36,24,24,36,66,0
4003 DATA 0,24,24,36,36,66,126,0
4004 DATA 0,126,64,124,64,64,126
,0
4005 DATA 0,16,56,84,84,56,16,0
4006 DATA 0,124,68,68,64,64,64,0
4007 DATA 0,66,66,126,66,66,66,0
4008 DATA 0,62,8,8,8,8,62,0
4009 DATA 0,0,0,0,0,0,0,0
4010 DATA 0,68,72,112,72,68,66,0
4011 DATA 0,24,36,66,66,66,66,0
4012 DATA 0,66,102,90,66,66,66,0
4013 DATA 0,66,98,82,74,70,66,0
4014 DATA 0,60,66,66,66,66,60,0
4015 DATA 0,126,36,36,36,36,36,0
4016 DATA 0,84,84,84,56,16,16,0
4017 DATA 0,124,66,66,124,64,64,
0
4018 DATA 0,124,32,16,16,32,124,
0
4019 DATA 0,254,16,16,16,16,16,0
4020 DATA 0,60,66,90,90,66,60,0
4021 DATA 0,0,0,0,0,0,0,0
4022 DATA 0,60,66,66,66,36,102,0
4023 DATA 0,0,124,0,56,0,124,0
4024 DATA 0,130,68,40,16,16,16,0
4025 DATA 0,126,4,8,16,32,126,0
4026 DATA 0,0,4,56,72,56,4,0
4027 DATA 0,56,36,56,36,56,32,0
4028 DATA 0,0,68,40,16,40,68,0
4029 DATA 0,8,16,8,24,36,24,0
4030 DATA 0,0,28,32,28,32,28,0
4031 DATA 0,0,16,56,84,56,16,0
4032 DATA 0,68,40,16,40,40,16,0
4033 DATA 0,0,120,66,66,66,4,0,
4034 DATA 0,0,16,16,16,20,24,0
4035 DATA 0,0,0,0,0,0,0,0
4036 DATA 0,32,40,48,48,40,36,0
4037 DATA 0,64,32,16,24,36,66,0
4038 DATA 0,0,36,36,36,56,64,0
4039 DATA 0,0,36,36,36,40,48,0
4040 DATA 0,0,56,68,68,68,56,0
4041 DATA 0,0,124,40,40,40,40,0
4042 DATA 0,0,16,84,84,56,16,0
4043 DATA 0,0,120,68,68,120,64,6
4
4044 DATA 0,0,0,62,72,72,48,0
4045 DATA 0,0,60,80,16,16,24,0
4046 DATA 0,0,24,36,60,36,24,0
4047 DATA 0,0,0,0,0,0,0,0
4048 DATA 0,0,36,66,66,90,36,0
4049 DATA 0,88,32,24,32,24,4,0
4050 DATA 0,0,36,84,20,20,24,0
4051 DATA 0,88,32,32,56,4,24,0
9000 POKE 23606,32: POKE 23607,1
38
```

CORRESPONDÊNCIA DE LETRAS

A e a — alpha	N e n — niú
B e b — beta	O e o — ómicron
C e c — qui	P e p — pi
D e d — delta	Q e q — psi
E e e — épsilon	R e r — ró
F e f — fi	S e s — sigma
G e g — gamma	T e t — tau
H e h — eta	U e u — teta
I e i — iota	V e v — espaço branco
J e j — espaço branco	W e w — omega
K e k — capa	X e x — csi
L e l — lambda	Y e y — iúpsilon
M e m — miú	Z e z — zeta

Neste programa: XXXXX = 35616 **XXXXX** = 35360 ni = 32 ns = 138

ANTÓNIO RIBEIRO

UTILITÁRIO

ZX SPECTRUM • TC 2048 • TC 2068

1250 ESC

O programa é um utilitário que transforma os bytes contidos nos endereços da RAM ou da ROM em caracteres ASCII de modo a dar a conhecer ao utilizador, os endereços onde estão as mensagens apresentadas por qualquer programa, e podendo até alterar os bytes, produzindo as suas próprias mensagens.

O programa é constituído por duas partes. A primeira, é responsável pela apresentação do menu de opções, que dará acesso à parte do programa em código máquina. O programa em código máquina começa no endereço 50000 e acaba em 50379.

Durante a execução, na parte inferior do ecrã está um contador que indica o endereço do caractere em destaque, no meio do ecrã. Para aumentar ou diminuir o endereço indicado, usam-se as teclas O — P. Todos os caracteres são deslocados pixel por pixel, para a esquerda ou para a direita, sendo possível ler as mensagens. Para parar o deslocamento, utiliza-se a tecla I, e para voltar ao menu, carrega-se em U.

Se o utilizador indicar a opção 4, e escrever o endereço 5010 (da ROM), poderá ver a localização das mensagens de erro do Spectrum.

PROGRAMAS

ALFREDO RODRIGUES

RENUMERAÇÃO

ZX SPECTRUM • TC 2048 • TC 2068

500 ESC

```
50233 205 124 196 .call 50300
50236 175 ..........xor a
50237 50 95 89 ....ld (22879),a
50240 195 212 195 .jp 50132
50243 14 8 ........ld c,8
50245 33 64 72 ....ld hl,18496
50248 6 8 .........ld b,8
50250 229 .........push hl
50251 197 .........push bc
50252 6 32 ........ld b,32
50254 203 30 .......rr (hl)
50256 44 ..........inc l
50257 16 251 ......djnz 50254
50259 193 .........pop bc
50260 225 .........pop hl
50261 36 ..........inc h
50262 167 .........and a
50263 16 241 ......djnz 50250
50265 118 .........halt
50266 13 ..........dec c
50267 32 232 ......jr nz,50245
50269 42 204 196 ..ld hl,(50380)
50272 43 ..........dec hl
50273 34 204 196 ..ld (50380),hl
50276 17 15 0 .....ld de,15
50279 237 82 ......sbc hl,de
50281 62 22 .......ld a,22
50283 215 .........rst 16
50284 62 10 .......ld a,10
50286 215 .........rst 16
50287 175 .........xor a
50288 215 .........rst 16
50289 205 124 196 .call 50300
50292 175 .........xor a
50293 50 81 89 ....ld (22865),a
50296 193 .........pop bc
50297 195 212 195 .jp 50132
50300 126 .........ld a,(hl)
50301 254 128 .....cp 128
50303 40 6 ........jr z,50311
50305 254 32 ......cp 32
50307 56 2 ........jr c,50311
50309 215 .........rst 16
50310 201 .........ret
50311 62 46 .......ld a,46
50313 215 .........rst 16
50314 201 .........ret
[illegible]
```

O programa é uma pequena rotina em Código Máquina, muito simples, que permite renumerar as linhas de um programa BASIC. A grande novidade desta não é a renumeração em si, mas sim a grande velocidade de execução comparada com um programa BASIC que faça o mesmo. Por outro lado para quem tiver, por exemplo, os comandos da impressora livres poderá fazer POKE na zona dos canais relativa à impressora do endereço onde foi introduzida a rotina, ficando os comandos da impressora LLIST e LPRINT a funcionar para esta rotina simulando o comando RENUMBER. Envio a listagem em assembler da rotina juntamente com os POKES a fazer na zona dos canais se o endereço onde foi introduzida a rotina for 60000.

Outra consideração que gostaria de fazer é que a rotina como está só permite renumerar de 10 em 10 começando no n.º 10 e se quisermos outra numeração teremos de fazer POKES nos endereços 60004 do byte - significativo e em 60005 do + significativo do número da primeira linha e em 60027 o - sig. e em 60028 o + sig. do valor do intervalo a deixar entre as linhas. Como exemplo se se fizer POKE 60004,5 ;POKE 60005,0; POKE 60027,10; POKE 60028,0 as linhas ficarão numeradas de 10 em 10 começando pelo n.º 5 se for executada a rotina. A rotina é relocatável e se executarmos com PRINT USR 60000 é imprimido no ecrã no fim da execução o número da última linha. De notar também que se usarmos LLIST para renumerar um programa BASIC já com algumas linhas demora muito mais tempo do que com LPRINT pelo que deve ser este o comando a usar se se optar por este processo.

```
10        LD HL,(PROG)   42 83 92
20        LD BC,NPL      1 XX XX
30 Rut    LD (HL),B      112
40        INC HL         35
50        LD (HL),C      113
60        INC HL         35
70        LD E,(HL)      94
80        INC HL         35
90        LD D,(HL)      86
100       ADD HL,DE      25
110       INC HL         35
120       PUSH HL        229
130       LD DE,(VARS)   237 91 75 92
140       AND A          167
150       SBC HL,DE      237 82
160       JR z Ret       40 10
170       AND A          167
180       LD HL,STEP     33 XX XX
190       ADD HL,BC      9
200       LD B,H         68
210       LD C,L         77
220       POP HL         225
230       JR Rut         24 227
240 Ret   POP HL         225
250       RET            201

NPL - Bytes - e + sig. do primeiro
numero de linha;

STEP - Bytes - e + sig. do interva
lo entre as linhas;

POKE 23749,XX (Byte - sig. do ende
reco de inicio da rotina;

POKE 23750,XX (Byte + sig. do ende
reco de inicio da rotina;

POKE 23749,96 e POKE 23750,234 para
o endereco 60000.
```

TIMEX

SISTEMA CP/M

TIMEX FDD 3000

— Sistema operativo CP/ (da Digital Research)
— Duas unidades de leitura Floppy
— 64 K de RAM

TIMEX TERMINAL

— Teclado Profissional 57 teclas
— Teclado numérico de 12 teclas

TIMEX PRINTER 2080

— Papel A-4 ou banda contínua
— 80 caracteres por linha (137 modo condensado)
— Tipos de letra (bold, elite, pica itálico, dupla largura, condensado, alta qualidade)

MONITOR

— 80 colunas
— 520 pontos por linha

Mais que um processador de texto, base de dados ou folha de cálculo,...

Um computador multilinguagem, basic, pascal, assembler...

SOFTWARE CP/M

— Flexiwrite (processador texto)
— Flexifile (base de dados)
— Flexicalc (folha de cálculo)

— Pascal 80 (linguagem)
— Devpac 80 (assembler)
— Basic (linguagem)

— Aplicações em contabilidade, stocks...